2015

Correctness results for on-line robust principal components analysis

Brian Thomas Lois
Iowa State University

Correctness results for on-line robust principal components analysis

by

Brian Thomas Lois

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Co-majors: Applied Mathematics

Electrical Engineering

Program of Study Committee:

Leslie Hogben, Co-major Professor

Namrata Vaswani, Co-major Professor

Nicola Elia

Fritz Keinert

Wolfgang Kliemann

Iowa State University

Ames, Iowa

2015

DEDICATION

I dedicate this thesis to Danielle. I am very grateful for all of her love and encouragement.

TABLE OF CONTENTS

LIST OF FIGURES vii

ACKNOWLEDGEMENTS ix

CHAPTER 1. INTRODUCTION 1

1.1 Thesis Organization 3

References 4

CHAPTER 2. ONLINE MATRIX COMPLETION AND ONLINE ROBUST PCA 6

2.1 Introduction 7

2.1.1 Problem Definition 8

2.1.2 Related Work 9

2.1.3 Contributions 10

2.1.4 Notation 11

2.1.5 Organization 12

2.2 Assumptions and Main Results 12

2.2.1 Model on $\boldsymbol{\ell}_t$ 13

2.2.2 Model on the set of missing entries or the outlier support set, $\mathcal{T}_t$ 15

2.2.3 Denseness 17

2.2.4 Main Result for Online Matrix Completion 17

2.2.5 Main Result for Online Robust PCA 20

2.2.6 Simple Generalizations 21

2.3 Discussion . . . 23
2.3.1 Discussion of the assumptions used . . . 23
2.3.2 Comparison with the results for PCP and NNM . . . 24
2.3.3 Other results for online RPCA and online MC . . . 25
2.4 Automatic ReProCS Algorithms for Online MC and Online RPCA and Why They Work . . . 27
2.4.1 Automatic ReProCS for Online MC (Algorithm 1) . . . 27
2.4.2 Automatic ReProCS for online RPCA (Algorithm 2) . . . 31
2.4.3 Key Insight for the Proof . . . 31
2.4.4 Proof Outline . . . 33
2.5 Most General Model on Changes in $\mathcal{T}_t$ and a Key Lemma . . . 34
2.5.1 Most General Model on Changes in $\mathcal{T}_t$ . . . 34
2.5.2 A Key Lemma that uses Model 2.5.1 . . . 35
2.6 Proof of Theorem 2.2.7 and Theorem 2.2.5 . . . 39
2.6.1 Definitions . . . 39
2.6.2 Five Main Lemmas for Proving Theorem 2.2.7 . . . 44
2.6.3 Proof of Theorem 2.2.7 . . . 46
2.6.4 Key Lemmas for Proving of Lemmas 2.6.16, 2.6.17, and 2.6.18 . . . 48
2.6.5 Proofs of Lemmas 2.6.16, 2.6.17, and 2.6.18 . . . 49
2.7 Proofs of Lemmas 2.6.21, 2.6.22, and 2.6.23 . . . 52
2.7.1 Some definitions, remarks and facts . . . 52
2.7.2 Preliminaries . . . 53
2.7.3 Simple Lemmas Needed for the Proofs . . . 54
2.7.4 Proofs of Lemma 2.6.21 and 2.6.22 . . . 55
2.7.5 Proof of Lemma 2.6.23 . . . 56
2.8 Simulation Experiments . . . 59

2.9 Extensions . . . 60
2.9.1 Other Models on Changes in $\mathcal{T}_t$ . . . 61
2.9.2 Analyze the ReProCS algorithm that also removes the deleted directions from the subspace estimate . . . 64
2.9.3 Relax the independence assumption on ℓ_t's . . . 64
2.9.4 Noisy and Undersampled Online Matrix Completion or Online Robust PCA . . . 65
2.10 Conclusions . . . 65
2.A Appendix A:
Proof that Model 2.2.3 on $\mathcal{T}_t$ satisfies the general Model 2.5.1 . . . 66
2.B Appendix B:
Proof of Lemma 2.6.14 (bound on $\zeta^+_{j,\text{new},k}$) and of Lemma 2.7.8 . . . 68
2.C Appendix C:
Proof of the Compressed Sensing (CS) Lemma (Lemma 2.6.15) . . . 70
2.D Appendix D: Proof of Cauchy-Schwarz inequality for matrices . . . 74
References . . . 74

CHAPTER 3. RECURSIVE SPARSE RECOVERY IN CORRELATED STRUCTURED NOISE . . . 77

3.1 Introduction . . . 77
3.1.1 Paper Organization . . . 78
3.1.2 Problem Definition . . . 79
3.1.3 Contribution . . . 79
3.1.4 Notation . . . 82
3.2 Model Assumptions, Main Result, and Discussion . . . 83
3.2.1 Model on $\boldsymbol{\ell}_t$ . . . 83
3.2.2 Denseness coefficient . . . 86
3.2.3 Model on $\boldsymbol{x}_t$ . . . 86
3.2.4 Main Result . . . 87
3.2.5 Random Support Change . . . 89

3.2.6 Discussion . . . 90
3.3 Most General Support Change Model and a Key Lemma . . . 95
3.3.1 Most General Support Change Model . . . 95
3.3.2 Key Lemma . . . 96
3.3.3 Unifying the signal models . . . 97
3.4 The Automatic ReProCS Algorithm . . . 98
3.5 Proof of Theorem 3.2.15 . . . 100
3.5.1 Definitions . . . 100
3.5.2 Main Lemmas . . . 102
3.5.3 Key Lemmas for Proving of Lemmas 3.5.11, 3.5.12, and 3.5.14 . . . 105
3.5.4 Proofs of Lemmas 3.5.11, 3.5.12, and 3.5.14 . . . 109
3.6 Proofs of Lemmas 3.5.20, 3.5.21, and 3.5.22 . . . 111
3.6.1 Key Lemmas Needed for the Proofs . . . 111
3.6.2 Proofs of Lemmas 3.5.20, 3.5.21, 3.5.22 . . . 114
3.7 Proof of Lemmas 3.6.1 and 3.6.3 . . . 120
3.8 Alternative Subspace Model and Algorithm . . . 123
3.8.1 Deletion Model . . . 123
3.8.2 Performance Guarantee for Algorithm 4 . . . 124
3.8.3 Discussion . . . 126
3.8.4 Cluster-PCA Algorithm (from [12]) . . . 127
3.8.5 Proof of Theorem 3.8.4 . . . 128
3.8.6 Proof of Lemma 3.8.8 . . . 132
3.9 Proofs of Lemmas 3.8.14, 3.8.15, and 3.8.16 . . . 134
3.9.1 Minor Lemmas for Proving the Main Lemmas . . . 134
3.10 Simulations . . . 142
3.A Preliminaries . . . 144
3.B Proofs of Support Change Lemmas . . . 151
3.C Bounding $\zeta^{+}_{k,\text{new}}$ [For the purposes of review] . . . 154
References . . . 155

CHAPTER 4. GENERAL CONCLUSIONS 158

References 158

LIST OF FIGURES

Figure 1.1 An fMRI sequence 2

Figure 2.1 A diagram of Model 2.2.2 13

Figure 2.3 $\varrho = 3$ and $\beta = 5$ case 16

Figure 2.4 $\varrho = 1$ and $\beta = 1$ case 16

Figure 2.5 Examples of Model 2.2.3. (a) shows a 1D video object of length s that moves by at least $s/3$ pixels once every 5 frames. (b) shows the object moving by s at every frame. (b) is an example of the best case for our result - the case with smallest ρ, β ($\mathcal{T}_t$'s mutually disjoint) 16

Figure 2.6 A diagram to visualize Algorithm 1 and Definition 2.6.4. The k-th projection-PCA step (at $t = \hat{t}_j + k\alpha$) computes the top left singular vectors of $(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')[\hat{\boldsymbol{\ell}}_{\hat{t}_j+(k-1)\alpha+1}, \hat{\boldsymbol{\ell}}_{\hat{t}_j+(k-1)\alpha+2}, \dots \hat{\boldsymbol{\ell}}_{\hat{t}_j+k\alpha}]$. . . . 41

Figure 2.7 Comparison of ReProCS and PCP for the RPCA problem. The top plot is the relative error $\|\boldsymbol{\ell}_t - \hat{\boldsymbol{\ell}}_t\|_2/\|\boldsymbol{\ell}_t\|_2$. The bottom plot shows the sparsity pattern of $\boldsymbol{S}$ (black represents a non-zero entry). Results are averaged over 100 simulations and plotted every 300 time instants. . . 61

Figure 2.8 Model 2.9.2 62

Figure 3.2 Signal Model 3.3.1 with $\varrho = 3$ 98

Figure 3.3 Disjoint Supports (Signal Model 3.3.1 with $\varrho = 1$) 98

Figure 3.4 Signal Model 3.2.12 98

Figure 3.5 Support Change Models 98

Figure 3.7 Signal Model 3.2.8 130

Figure 3.8 Algorithm 4 130

Figure 3.9 Diagrams for Signal Model 3.2.8 and Algorithm 4 130

Figure 3.10 Support of $\boldsymbol{X}$ determined by Bernoulli model. The y axis is $\frac{\|\hat{\boldsymbol{x}}_t - \boldsymbol{x}_t\|_2}{\|\boldsymbol{x}_t\|_2}$. . 143

Figure 3.11 Support of $\boldsymbol{X}$ obeys Signal Model 3.2.11. The y axis is $\frac{\|\hat{\boldsymbol{x}}_t - \boldsymbol{x}_t\|_2}{\|\boldsymbol{x}_t\|_2}$. . . . 144

ACKNOWLEDGEMENTS

I would like to thank Dr. Leslie Hogben for sharing her wisdom with me throughout my graduate career. Leslie has always been supportive and encouraging, helping me to achieve my goals. I would also like to thank Dr. Namrata Vaswani for her guidance and contributions to this research. Thank you also to my committee members, Dr. Nicola Elia, Dr. Fritz Keinert, and Dr. Wolfgang Kliemann, for their time and support. Finally, thank you to all of my family, friends, and fellow graduate students who have helped me complete this work.

CHAPTER 1. INTRODUCTION

This thesis considers the problem of identifying two vectors from knowledge of their sum. Obviously this is impossible in general. However, when a time series of vectors is available and some structure is assumed on both vectors, identification becomes possible. Th structure assumed here is that one vector is sparse, that is, most of its entries are zero, and the other vector lies in a low-dimensional subspace, that is, if the vectors in the time series are horizontally concatenated they will form a low-rank matrix. Suppose that at each time t a vector $\boldsymbol{m}_t$ is observed where

$$\boldsymbol{m}_t = \boldsymbol{x}_t + \boldsymbol{\ell}_t$$

with $\boldsymbol{x}_t$ sparse and the $\boldsymbol{\ell}_t$ all belonging to some low-dimensional subspace. The goal is to recover $\boldsymbol{x}_t$ and $\boldsymbol{\ell}_t$ at each time t.

Even with the assumption that the $\boldsymbol{x}_t$ are sparse, and the $\boldsymbol{\ell}_t$ lie in a low-dimensional subspace, there is still an identifiability problem. Sparse vectors when horizontally concatenated could form a low rank matrix. Similarly, vectors in the low-dimensional subspace could be sparse. So it is also assumed that the $\boldsymbol{x}_t$ are not low-dimensional and the $\boldsymbol{\ell}_t$ are not sparse. The property of not being sparse is called denseness, and is quantified in the sequel. Rather than directly assuming that $\boldsymbol{x}_t$ are not low-rank, the results require that the supports of $\boldsymbol{x}_t$ for distinct t be sufficiently different.

As a motivation for our problem, imagine video sequence with a distinct background and foreground. Suppose for example that a fixed surveillance camera records a scene as a person walks through. The background images, which do not change very much, can be modeled as belonging to a low-dimensional subspace. The foreground (person) is small compared to the size of the image, so can be modeled as a sparse vector.

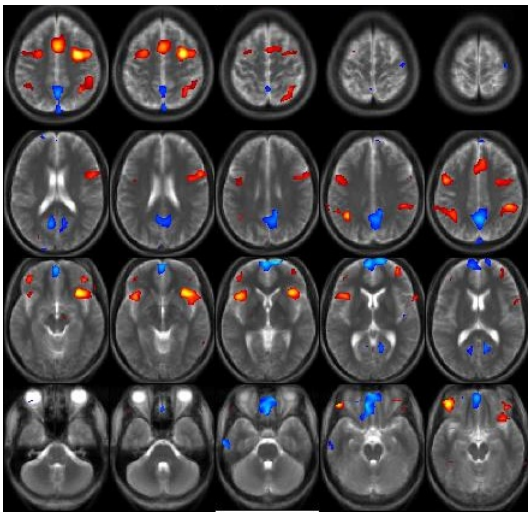

Figure 1.1: An fMRI sequence

A similar application of background/foreground separation arises in functional magnetic resonance imaging (fMRI). During an fMRI procedure, images of a patient's brain are produced that show which areas are "active" at a given time. This can be used to map out which areas of the brain are used for different tasks. One can see from Figure 1.1 that the backgrounds from left to right remain largely similar, and the part that is lit up only covers a small amount of the overall image.

The above problem can be interpreted in two different ways. If the $\boldsymbol{x}_t$ are of primary interest, the problem is one of sparse recovery. Recently, much work has been done on the problem of recovering sparse vectors. An example of early work is [1], which demonstrates the effectiveness of the ℓ_1 norm for sparse reconstruction. In [2], the author considers the following problem. Let $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}$ where x is sparse and $\boldsymbol{b}$ is small noise. That is $\|\boldsymbol{b}\|_2 < \epsilon$ for some $\epsilon > 0$. The problem is to recover $\boldsymbol{x}$ from $\boldsymbol{y}$. It is proved that under certain conditions on the matrix $\boldsymbol{A}$, the following convex program will recover $\boldsymbol{x}$ with error bounded by a small constant times ϵ:

$$\min_{\tilde{\boldsymbol{x}}} \|\tilde{\boldsymbol{x}}\|_1 \text{ subject to } \|\boldsymbol{y} - \boldsymbol{A}\tilde{\boldsymbol{x}}\|_2 < \epsilon. \tag{1.1}$$

This procedure will be used as part of a larger algorithm to solve the original problem of separating $\boldsymbol{x}_t$ and $\boldsymbol{\ell}_t$. The reason (1.1) alone will not work here is that $\boldsymbol{\ell}_t$ is not necessarily small in magnitude.

Another interpretation of the original problem is as a robust principal components analysis (PCA) problem. Given a matrix of data, the goal of PCA is to compute a small number or orthogonal directions along which most of the variability in the columns lie. Because traditional procedures for PCA are sensitive to outliers in the data, much work has been done to develop algorithms for PCA that are robust with respect to outliers. In seminal work, Candès et al. [3] and Chandrasekaran et al. [4] both posed the robust PCA problem as separating a sparse matrix $\boldsymbol{X}$ from a low-rank matrix $\boldsymbol{L}$ from knowledge of their sum $\boldsymbol{M} = \boldsymbol{X} + \boldsymbol{L}$. Each proved that the convex program,

$$\min_{\tilde{\boldsymbol{X}},\tilde{\boldsymbol{L}}} \|\tilde{\boldsymbol{X}}\|_{\text{sum}} + \gamma\|\tilde{\boldsymbol{L}}\|_* \text{ subject to } \tilde{\boldsymbol{X}} + \tilde{\boldsymbol{L}} = \boldsymbol{M} \tag{1.2}$$

will exactly recover $\boldsymbol{X}$ and $\boldsymbol{L}$ under certain (fairly mild) conditions. In the above, γ is a scalar parameter, $\|\cdot\|_{\text{sum}}$ is the vector ℓ_1 norm of a matrix (sum of absolute values of entries), and $\|\cdot\|_*$ is the nuclear norm (sum of the singular values). The problem studied in this thesis can be viewed as a recursive or on-line version of the robust PCA problem as posed in [3] and [4]. In this thesis, the goal is to recover the columns of $\boldsymbol{X}$ and $\boldsymbol{L}$ as they arrive. One approach would be to re-solve (1.2) each time a new column arrives, but this is not computationally feasible when a real-time solution is desired.

This thesis analyzes versions of an algorithm called recursive projected compressed sensing (ReProCS). At a high level, ReProCS works as follows: if an accurate estimate of the subspace where the $\boldsymbol{\ell}_t$'s lie is available, then projecting perpendicular to this subspace estimate will nullify most of $\boldsymbol{\ell}_t$. Let $\boldsymbol{\Phi}_t$ denote this projection. Then $\boldsymbol{\Phi}_t\boldsymbol{m}_t = \boldsymbol{\Phi}_t\boldsymbol{x}_t + \boldsymbol{\Phi}_t\boldsymbol{\ell}_t$ where $\|\boldsymbol{\Phi}_t\boldsymbol{\ell}_t\|_2$ is small. Finding $\boldsymbol{x}_t$ is now a sparse recovery problem in small noise, so (1.1) can be used to accurately recover $\boldsymbol{x}_t$. By subtracting the estimate of $\boldsymbol{x}_t$ from $\boldsymbol{m}_t$, an estimate of $\boldsymbol{\ell}_t$ is obtained. Finally, the estimates of $\boldsymbol{\ell}_t$ are used to maintain an accurate estimate of the subspace where the $\boldsymbol{\ell}_t$s lie as the subspace changes slowly over time.

1.1 Thesis Organization

The format of this thesis is journal papers in a thesis. Chapter 1 introduces the problem studied and gives some background information.

Chapter 2 contains the paper "Online Matrix Completion and Online Robust PCA," a version of which has been submitted to *IEEE Transactions on Information Theory* and is under review. Minor modifications have been made to the submitted version for the purposes of this thesis. This paper proves a complete correctness for the ReProCS algorithm under four main assumptions: 1) an accurate estimate of the initial subspace is available; 2) the vectors $\boldsymbol{\ell}_t$ are mutually independent over time, dense, and the subspace where they lie changes slowly; 3) the support of $\boldsymbol{x}_t$ changes 'enough' over time; 4) the algorithm parameters are set appropriately. All of these assumptions are made precise and quantified in the paper itself. I was the primary author and researcher for this paper. The overall structure of the proof follows that in [5]. The pieces needed to obtain a complete correctness result were proved by me. Namrata Vaswani contributed both research and writing.

Chapter 3 contains the paper "Recursive Sparse Recovery in Correlated Structured Noise," a paper that is being prepared for submission to *IEEE Transactions on Information Theory*. In ongoing work, revisions have been made, and new results have been added to the paper. The paper in this thesis contains two important improvements over the previous paper. First, the assumption that the $\boldsymbol{\ell}_t$s are independent over time is relaxed, and a first order autoregressive model is used for the coefficients of $\boldsymbol{\ell}_t$. Second, a subspace update step is added to the ReProCS algorithm. This update step allows for old directions to be deleted from the estimate of the subspace, which in turn allows for a relaxed denseness assumption on the low-dimensional vectors. I was the primary author of this paper and proved the complete correctness result. Jinchun Zhan did initial research on the correlated model, and Namrata Vaswani contributed research and writing.

Chapter 4 gives general conclusions and directions for future research.

References

[1] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal of Scientific Computing*, vol. 20, pp. 33–61, 1998.

[2] E. Candes, "The restricted isometry property and its implications for compressed sensing," *Compte Rendus de l'Academie des Sciences, Paris, Serie I*, pp. 589–592, 2008.

[3] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of ACM*, vol. 58, no. 3, 2011.

[4] V. Chandrasekaran, S. Sanghavi, P. A. Parrilo, and A. S. Willsky, "Rank-sparsity incoherence for matrix decomposition," *SIAM Journal on Optimization*, vol. 21, 2011.

[5] C. Qiu, N. Vaswani, B. Lois, and L. Hogben, "Recursive robust pca or recursive sparse recovery in large but structured noise," *IEEE Trans. Info. Th.*, Aug. 2014, shorter versions in ICASSP 2013 and ISIT 2013.

CHAPTER 2. ONLINE MATRIX COMPLETION AND ONLINE ROBUST PCA

A paper submitted to *IEEE Transactions on Information Theory*

Brian Lois and Namrata Vaswani

Abstract

This work studies two interrelated problems - online robust PCA (RPCA) and online low-rank matrix completion (MC). In recent work by Candès et al., RPCA has been defined as a problem of separating a low-rank matrix (true data), $L := [\ell_1, \ell_2, \dots \ell_t, \dots, \ell_{t_{\max}}]$ and a sparse matrix (outliers), $S := [x_1, x_2, \dots x_t, \dots, x_{t_{\max}}]$ from their sum, $M := L + S$. Our work uses this definition of RPCA. An important application where both these problems occur is in video analytics in trying to separate sparse foregrounds (e.g., moving objects) and slowly changing backgrounds.

While there has been a large amount of recent work on both developing and analyzing batch RPCA and batch MC algorithms, the online problem is largely open. In this work, we develop a practical modification of our recently proposed algorithm to solve both the online RPCA and online MC problems. The main contribution of this work is that we obtain correctness results for the proposed algorithms under mild assumptions. The assumptions that we need are: (a) a good estimate of the initial subspace is available (easy to obtain using a short sequence of background-only frames in video surveillance); (b) the ℓ_t's obey a 'slow subspace change' assumption; (c) the basis vectors for the subspace from which ℓ_t is generated are dense (non-sparse); (d) the support of x_t changes by at least a certain amount at least every so often; and (e) algorithm parameters are appropriately set.

2.1 Introduction

Principal Components Analysis (PCA) is a tool that is frequently used for dimension reduction. Given a matrix of data $\boldsymbol{D}$, PCA computes a small number of orthogonal directions, called principal components, that contain most of the variability of the data. For relatively noise-free data that lies close to a low-dimensional subspace, PCA is easily accomplished via singular value decomposition (SVD). The problem of PCA in the presence of outliers is referred to as robust PCA (RPCA). In recent work, Candès et al. [1] posed RPCA as a problem of separating a low-rank matrix, $\boldsymbol{L}$, and a sparse matrix, $\boldsymbol{S}$, from their sum, $\boldsymbol{M} := \boldsymbol{L} + \boldsymbol{S}$. They proposed a convex program called principal components' pursuit (PCP) that provided a provably correct batch solution to this problem under mild assumptions. PCP solves

$$\min_{\tilde{\boldsymbol{L}},\tilde{\boldsymbol{S}}} \|\tilde{\boldsymbol{L}}\|_* + \lambda\|\tilde{\boldsymbol{S}}\|_{\text{sum}} \quad \text{subject to} \quad \tilde{\boldsymbol{L}} + \tilde{\boldsymbol{S}} = \boldsymbol{M},$$

where $\|\cdot\|_*$ is the nuclear norm (sum of singular values), $\|\cdot\|_{\text{sum}}$ is the sum of the absolute values of the entries, and λ is an appropriately chosen scalar. The same program was analyzed in parallel by Chandrasekharan et al. [2] and later by Hsu et al. [3]. Since these works, there has been a large amount of work on batch approaches for RPCA and their performance guarantees.

When RPCA needs to be solved in a recursive fashion for sequentially arriving data vectors it is referred to as online (or recursive) RPCA. Online RPCA assumes that a short sequence of outlier-free (sparse component free) data vectors is available. An example application where this problem occurs is the problem of separating a video sequence into foreground and background layers (video layering) on-the-fly [1]. Video layering is a key first step for automatic video surveillance and many other streaming video analytics tasks. In videos, the foreground usually consists of one or more moving persons or objects and hence is a sparse image. The background images usually change only gradually over time [1], e.g., moving lake waters or moving trees in a forest, and hence are well modeled as lying in a low-dimensional subspace that is fixed or slowly changing. Also, the changes are global (dense) [1]. In most video applications, it is valid to assume that an initial short sequence of background-only frames is available and this can be used to estimate the initial subspace via SVD.

Often in video applications the sparse foreground $\boldsymbol{x}_t$ is actually the signal of interest, and the background $\boldsymbol{\ell}_t$ is the noise. In this case, the problem can be interpreted as one of recursive sparse recovery in (potentially) large but structured noise. Our result allows for $\boldsymbol{\ell}_t$ to be large in magnitude as long as it is structured. The structure we impose is that the $\boldsymbol{\ell}_t$'s lie in a low dimensional subspace that changes *slowly* over time.

In some other applications, instead of there being outliers, parts of a data vector may be missing entirely. When the (unknown) complete data vector is a column of a low-rank matrix, the problem of recovering it is referred to as matrix completion (MC). For example, recovering video sequences and tracking their subspace changes in the presence of easily detectable foreground occlusions. If the occluding object's intensity is known and is significantly different from that of the background, its support can be obtained by simple thresholding. The background video recovery problem then becomes an MC problem. A nuclear norm minimization (NNM) based solution for MC was introduced in [4] and studied in [5]. The convex program here is to minimize the nuclear norm of $\tilde{\boldsymbol{M}}$ subject to $\tilde{\boldsymbol{M}}$ and $\boldsymbol{M}$ agreeing on all observed entries. Since then there has been a large amount of work on batch methods for MC and their correctness results.

2.1.1 Problem Definition

Consider the *online MC problem.* Let $\mathcal{T}_t$ denote the set of missing entries at time t. We observe a vector $\boldsymbol{m}_t \in \mathbb{R}^n$ that satisfies

$$\boldsymbol{m}_t = \boldsymbol{I}_{\overline{\mathcal{T}}_t} \boldsymbol{I}_{\overline{\mathcal{T}}_t}{}' \boldsymbol{\ell}_t \qquad \text{for } t = t_{\text{train}} + 1, t_{\text{train}} + 2, \ldots, t_{\max}. \tag{2.1}$$

with the possibility that $t_{\max}$ can be infinity. Here $\boldsymbol{\ell}_t$ is such that, for t large enough (quantified in Model 2.2.2), the matrix $\boldsymbol{L}_t := [\boldsymbol{\ell}_1, \boldsymbol{\ell}_2, \ldots, \boldsymbol{\ell}_t]$ is rank deficient. Notice that by defining $\boldsymbol{m}_t$ as above, we are setting to zero the entries that are missed (see the notation section on page 11).

Consider the *online RPCA* problem. At time t we observe a vector $\boldsymbol{m}_t \in \mathbb{R}^n$ that satisfies

$$\boldsymbol{m}_t = \boldsymbol{\ell}_t + \boldsymbol{x}_t \qquad \text{for } t = t_{\text{train}} + 1, t_{\text{train}} + 2, \ldots, t_{\max}. \tag{2.2}$$

Here $\boldsymbol{\ell}_t$ is as defined above and $\boldsymbol{x}_t$ is the sparse (outlier) vector. We use $\mathcal{T}_t$ to denote the support set of $\boldsymbol{x}_t$.

For both problems above, for $t = 1, 2, \ldots, t_{\text{train}}$, we are given complete outlier-free measurements $\boldsymbol{m}_t = \boldsymbol{\ell}_t$ so that it is possible to estimate the initial subspace. For the video surveillance application, this would correspond to having a short initial sequence of background only images, which can often be obtained. For $t > t_{\text{train}}$, the goal is to estimate $\boldsymbol{\ell}_t$ (or $\boldsymbol{\ell}_t$ and $\boldsymbol{x}_t$ in case of RPCA) as soon as $\boldsymbol{m}_t$ arrives and to periodically update the estimate of range($\boldsymbol{L}_t$).

In the rest of the paper, we refer to $\mathcal{T}_t$ as the *missing/corrupted entries set*.

2.1.2 Related Work

Some other work that also studies the online MC problem (defined differently from above) includes [6, 7, 8, 9]. We discuss the connection with the idea from [6] in Section 2.4. The algorithm from [7], GROUSE, is a first order stochastic gradient method; a result for its convergence to the local minimum of the cost function it optimizes is obtained in [9]. The algorithm of [8], PETRELS, is a second order stochastic gradient method. It is shown in [8] that PETRELS converges to the stationary point of the cost function it optimizes. The advantage of PETRELS and GROUSE is that they do not need initial subspace knowledge. Another somewhat related work is [10].

Partial results have been provided for ReProCS for online RPCA in our older work [11]. In other more recent work [12] another partial result is obtained for online RPCA defined differently from above. Neither of these is a correctness result. Both require an assumption that depends on intermediate algorithm estimates. Another somewhat related work is [13] on online PCA with contaminated data. This does not model the outlier as a sparse vector but defines anything that is far from the data subspace as an outlier.

Some other works only provide an algorithm without proving any performance results, e.g., [14].

We discuss the most related works in detail in Sec 2.3.3.

2.1.3 Contributions

In this work we develop and study a practical modification of the Recursive Projected Compressive Sensing (ReProCS) algorithm introduced and studied in our earlier work [11] for online RPCA. We also develop a special case of it that solves the online MC problem. The main contribution of this work is that we obtain a *complete correctness result* for ReProCS-based algorithms for both online MC and online RPCA (or more generally, online sparse plus low-rank matrix recovery). Online algorithms are useful because they are causal (needed for applications like video surveillance) and, in most cases, are faster and need less storage compared to most batch techniques (we should mention here that there is some recent work on faster batch techniques as well, e.g., [15]). To the best of our knowledge, this work and an earlier conference version of this [16] may be among the first correctness results for online RPCA. The algorithm studied in [16] is more restrictive.

Moreover, as we will see, by exploiting temporal dependencies, such as slow subspace change, and initial subspace knowledge, our result is able to allow for a more correlated set of missing/corrupted entries than do the various results for PCP [1, 2, 3] or NNM [5] (see Sec. 2.3).

Our result uses the overall proof approach introduced in our earlier work [11] that provided a partial result for online RPCA. The most important new insight needed to get a complete result is described in Section 2.4.3. Also see Sec. 2.3.3. New proof techniques are needed for this line of work because almost all existing works only analyze batch algorithms that solve a different problem. Also, as explained in Section 2.4, the standard PCA procedure cannot be used in the subspace update step and hence results for it are not applicable.

As shown in [17], because it exploits initial subspace knowledge and slow subspace change, ReProCS has significantly improved recovery performance compared with batch RPCA algorithms - PCP [1] and [18] - as well as with the online algorithm of [14] for foreground and background extraction in many simulated and real video sequences; it is also faster than the batch methods but slower than [14].

2.1.4 Notation

We use $'$ to denote transpose. The 2-norm of a vector and the induced 2-norm of a matrix are denoted by $\|\cdot\|_2$. For a set $\mathcal{T}$ of integers, $|\mathcal{T}|$ denotes its cardinality and $\overline{\mathcal{T}}$ denotes its complement set. We use $\emptyset$ to denote the empty set. For a vector $\boldsymbol{x}$, $\boldsymbol{x}_{\mathcal{T}}$ is a smaller vector containing the entries of $\boldsymbol{x}$ indexed by $\mathcal{T}$. Define $\boldsymbol{I}_{\mathcal{T}}$ to be an $n \times |\mathcal{T}|$ matrix of those columns of the identity matrix indexed by $\mathcal{T}$. For a matrix $\boldsymbol{A}$, define $\boldsymbol{A}_{\mathcal{T}} := \boldsymbol{A}\boldsymbol{I}_{\mathcal{T}}$. For matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ where the columns of $\boldsymbol{Q}$ are a subset of the columns of $\boldsymbol{P}$, $\boldsymbol{P} \setminus \boldsymbol{Q}$ refers to the matrix of columns in $\boldsymbol{P}$ and not in $\boldsymbol{Q}$.

For an $n \times n$ Hermitian matrix $\boldsymbol{H}$, $\boldsymbol{H} \overset{\text{EVD}}{=} \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}'$ denotes an eigenvalue decomposition. That is, $\boldsymbol{U}$ has orthonormal columns, and $\boldsymbol{\Lambda}$ is a diagonal matrix of size at least $\text{rank}(\boldsymbol{H}) \times \text{rank}(\boldsymbol{H})$. (If $\boldsymbol{H}$ is rank deficient, then $\boldsymbol{\Lambda}$ can have any size between $\text{rank}(\boldsymbol{H})$ and n.) For Hermitian matrices $\boldsymbol{A}$ and $\boldsymbol{B}$, the notation $\boldsymbol{A} \preceq \boldsymbol{B}$ means that $\boldsymbol{B} - \boldsymbol{A}$ is positive semi-definite. We order the eigenvalues of an Hermitian matrix in decreasing order. So $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$.

For integers a and b, we use the interval notation $[a, b]$ to mean all of the integers between a and b, inclusive, and similarly for $[a, b)$ etc.

Definition 2.1.1. *For a matrix $\boldsymbol{A}$, the restricted isometry constant (RIC) $\delta_s(\boldsymbol{A})$ is the smallest real number δ_s such that*

$$(1 - \delta_s)\|\boldsymbol{x}\|_2^2 \leq \|\boldsymbol{A}\boldsymbol{x}\|_2^2 \leq (1 + \delta_s)\|\boldsymbol{x}\|_2^2$$

for all s-sparse vectors $\boldsymbol{x}$ [19]. A vector $\boldsymbol{x}$ is s-sparse if it has s or fewer non-zero entries.

Definition 2.1.2. *We refer to a matrix with orthonormal columns as a* basis matrix*. Notice that if $\boldsymbol{P}$ is a basis matrix, then $\boldsymbol{P}'\boldsymbol{P} = \boldsymbol{I}$.*

Definition 2.1.3. *For basis matrices $\hat{\boldsymbol{P}}$ and $\boldsymbol{P}$, define* $\text{dif}(\hat{\boldsymbol{P}}, \boldsymbol{P}) := \|(\boldsymbol{I} - \hat{\boldsymbol{P}}\hat{\boldsymbol{P}}')\boldsymbol{P}\|_2$. *This quantifies the difference between their range spaces.*

If $\hat{\boldsymbol{P}}$ and $\boldsymbol{P}$ have the same number of columns, then $\text{dif}(\hat{\boldsymbol{P}}, \boldsymbol{P}) = \text{dif}(\boldsymbol{P}, \hat{\boldsymbol{P}})$, otherwise the function is not necessarily symmetric.

2.1.5 Organization

The remainder of the paper is organized as follows. In Section 2.2 we give the model and main result for both online MC and online RPCA. Next we discuss our main results in Section 2.3. The algorithms for solving both problems are given and discussed in Section 2.4. The discussion also explains why the proof of our main result should go through. Section 2.4.3 within this section describes the key insight needed by the proof and Section 2.4.4 gives the proof outline. The most general form of our model on the missing entries set, $\mathcal{T}_t$, is described in Section 2.5. A key new lemma for proving our main results is also given in this section. The proof of our main results can be found in Section 2.6. Proofs of three long lemmas needed for proving the lemmas leading to the main theorem are postponed until Section 2.7. Section 2.8 shows numerical experiments backing up our claims. We discuss some extensions in Section 2.9 and give conclusions in Section 2.10

2.2 Assumptions and Main Results

Before we give our model on $\boldsymbol{\ell}_t$, we need the following definition.

Definition 2.2.1. *Recall that $\boldsymbol{m}_t = \boldsymbol{\ell}_t$ for $t = 1, \ldots, t_{\text{train}}$ is the training data. Let $\hat{\lambda}^-_{\text{train}}$ be the minimum non-zero eigenvalue of $\frac{1}{t_{\text{train}}} \sum_{t=1}^{t_{\text{train}}} \boldsymbol{m}_t \boldsymbol{m}_t{}'$. That is*

$$\hat{\lambda}^-_{\text{train}} := \min_{\lambda_i > 0} \lambda_i \left(\frac{1}{t_{\text{train}}} \sum_{t=1}^{t_{\text{train}}} \boldsymbol{m}_t \boldsymbol{m}_t{}' \right)$$

Define $\hat{\boldsymbol{P}}_{t_{\text{train}}}$ to be the matrix containing the eigenvectors of $\frac{1}{t_{\text{train}}} \sum_{t=1}^{t_{\text{train}}} \boldsymbol{m}_t \boldsymbol{m}_t{}'$, with eigenvalues larger than or equal to $\hat{\lambda}^-_{\text{train}}$, as its columns.

We will use $\hat{\lambda}^-_{\text{train}}$ in our algorithms to set the eigenvalue threshold to both detect subspace change and estimate the number of newly added directions. We also use $\hat{\lambda}^-_{\text{train}}$ to state the slow subspace change assumption below. We will use $\hat{\boldsymbol{P}}_{t_{\text{train}}}$ as the initial subspace knowledge in the algorithms.

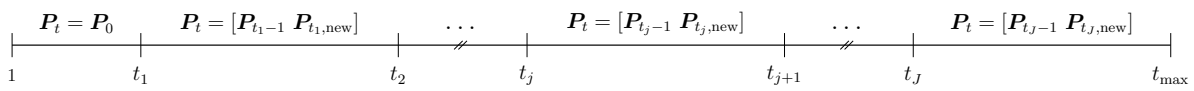

Figure 2.1: A diagram of Model 2.2.2

2.2.1 Model on $\boldsymbol{\ell}_t$

We assume that $\boldsymbol{\ell}_t$ is a vector from a slowly changing low-dimensional subspace that changes in such a way that the matrix $\boldsymbol{L}_t := [\boldsymbol{\ell}_1, \boldsymbol{\ell}_2, \dots \boldsymbol{\ell}_t]$ is low-rank for t large enough. This can be modeled in various ways. One possible model is given below. It assumes that $\boldsymbol{\ell}_t$'s are zero mean, bounded and mutually independent random variables with a covariance matrix that is low-rank at each time and that changes "slowly" in the following fashion: (a) its column subspace remains constant for a long enough time and then changes; (b) when it changes, the number of newly added directions is small and the eigenvalues along the newly added directions are small for d frames after the change.

Model 2.2.2 (Model on $\boldsymbol{\ell}_t$)**.** *Assume that the $\boldsymbol{\ell}_t$ are zero mean and bounded random vectors in $\mathbb{R}^n$ that are mutually independent over time. Also assume that their covariance matrix $\boldsymbol{\Sigma}_t$ has an eigenvalue decomposition*

$$\mathbb{E}[\boldsymbol{\ell}_t \boldsymbol{\ell}_t{}'] = \boldsymbol{\Sigma}_t \overset{\text{EVD}}{=} \boldsymbol{P}_t \boldsymbol{\Lambda}_t \boldsymbol{P}_t{}'$$

where $\boldsymbol{P}_t$ changes as

$$\boldsymbol{P}_t = \begin{cases} [\boldsymbol{P}_{t-1}\ \boldsymbol{P}_{t,\text{new}}] & \text{if } t = t_1 \text{ or } t_2 \text{ or } \dots \ t_J \\ \boldsymbol{P}_{t-1} & \text{otherwise.} \end{cases} \tag{2.3}$$

and $\boldsymbol{\Lambda}_t$ changes as follows. For $t \in [t_j, t_{j+1})$, define $\boldsymbol{\Lambda}_{t,\text{new}} := \boldsymbol{P}_{t_j,\text{new}}{}' \boldsymbol{\Sigma}_t \boldsymbol{P}_{t_j,\text{new}}$ and assume that

$$(\boldsymbol{\Lambda}_{t,\text{new}})_{i,i} = (v_i)^{t-t_j} q_i \hat{\lambda}^-_{\text{train}} \quad \text{for} \quad i = 1, \dots, r_{j,\text{new}} \tag{2.4}$$

where $q_i \geq 1$ and $v_i > 1$ but not too large [1]*. We assume that (a) $t_{j+1} - t_j \geq d$ for a $d \geq (K+2)\alpha$;*

[1] *Our result would still hold if the v_i were different for each change time (i.e. $v_{j,i}$). We let them be the same to reduce notation.*

and (b) for all i, $q_i(v_i)^d \leq 3$. Here K and α are algorithm parameters that are set in Theorem 2.2.7.

Other minor assumptions are as follows. (i) Define $t_0 := 1$ and assume that $t_{\text{train}} \in [t_0, t_1)$. (ii) For $j = 0, 1, 2, \ldots, J$, define

$$r_j := \text{rank}(\boldsymbol{P}_{t_j}) \quad \textit{and} \quad r_{j,\text{new}} := \text{rank}(\boldsymbol{P}_{t_j,\text{new}}).$$

and assume that $r_J < \min(n, t_{j+1} - t_j)$. This ensures that, for all $t > r_J$, the matrix $\boldsymbol{L}_t$ is low-rank. (iii) Define

$$\lambda^+ := \sup_t \lambda_{\max}(\boldsymbol{\Lambda}_t)$$

as the maximum eigenvalue at any time and assume that $\lambda^+ < \infty$.

Observe from the above that $\boldsymbol{P}_t$ is a basis matrix and $\boldsymbol{\Lambda}_t$ is diagonal. We refer to the t_j's as the subspace change times.

A visual depiction of the above model can be found in Figure 2.1.

Define the largest and smallest eigenvalues along the new directions for the first d frames after a subspace change as

$$\lambda^+_{\text{new}} := \max_j \max_{t \in [t_j, t_j+d]} \lambda_{\max}\left(\boldsymbol{\Lambda}_{t,\text{new}}\right) \text{ and } \lambda^-_{\text{new}} := \min_j \min_{t \in [t_j, t_j+d]} \lambda_{\min}\left(\boldsymbol{\Lambda}_{t,\text{new}}\right)$$

The slow change model on $\boldsymbol{\Lambda}_{t,\text{new}}$ is one way to ensure that

$$\hat{\lambda}^-_{\text{train}} \leq \lambda^-_{\text{new}} \leq \lambda^+_{\text{new}} \leq 3\hat{\lambda}^-_{\text{train}} \tag{2.5}$$

i.e. the maximum variance of the projection of $\boldsymbol{\ell}_t$ along the new directions is small enough for the first d frames after a change. Also the minimum variance is larger than a constant greater than zero (and hence detectable). The proof of our main result only relies on (2.5) and does not use the actual slow increase model in any other way. The above inequality along with $t_{j+1} - t_j \geq d \geq (K+2)\alpha$ quantifies *"slow subspace change"*.

Notice that the above model does not put any assumption on the eigenvalues along the existing directions. In particular, they do not need to be greater than zero and hence the model automatically allows existing directions (columns of $\boldsymbol{P}_{t_j-1}$ for $t \in [t_j, t_{j+1})$) to drop out of the current subspace. It could be the case that for some time period, $(\boldsymbol{\Lambda}_t)_{i,i} = 0$ (for an

i corresponding to a column of $\boldsymbol{P}_{t_j-1}$), so that the i^{th} column of $\boldsymbol{P}_{t_j-1}$ is not contributing anything to $\boldsymbol{\ell}_t$ at that time. For the same index i, $(\boldsymbol{\Lambda}_t)_{i,i}$ could also later increase again to a nonzero value. Therefore $r_0 + \sum_{i=1}^{j} r_{i,\text{new}}$ is only a bound on the rank of $\boldsymbol{\Sigma}_t$ for $t \in [t_j, t_{j+1})$, and not necessarily the rank itself. A more explicit model for deletion of directions is to let $\boldsymbol{P}_t$ change as

$$\boldsymbol{P}_t = \begin{cases} [(\boldsymbol{P}_{t-1} \setminus \boldsymbol{P}_{t,\text{del}}) \;\; \boldsymbol{P}_{t,\text{new}}] & \text{if } t = t_1 \text{ or } t_2 \text{ or } \ldots \; t_J \\ \boldsymbol{P}_{t-1} & \text{otherwise.} \end{cases} \tag{2.6}$$

where $\boldsymbol{P}_{t,\text{del}}$ contains the columns of $\boldsymbol{P}_{t-1}$ for which the variance is zero. If we add the assumption that $[\boldsymbol{P}_{t_1-1} \; \boldsymbol{P}_{t_1,\text{new}} \; \boldsymbol{P}_{t_2,\text{new}} \; \ldots \; \boldsymbol{P}_{t_J,\text{new}}]$ be a basis matrix (i.e. deleted directions cannot be part of a later $\boldsymbol{P}_{t_j,\text{new}}$), then this is a special case of Model 2.2.2 above. We say special case because this only allows deletions at times t_j, whereas Model 2.2.2 allows deletion of old directions at any time.

For $t \in [t_j, t_{j+1})$, let $\boldsymbol{P}_{t,*} := \boldsymbol{P}_{t_j-1}$ and $\boldsymbol{\Lambda}_{t,*} := \boldsymbol{P}_{t,*}{}'\boldsymbol{\Sigma}_t\boldsymbol{P}_{t,*}$. Observe that Model 2.2.2 does not have any constraint on $\boldsymbol{\Lambda}_{t,*}$. Thus if we assume that its entries are such that their changes from t to $t+1$ are smaller than or equal to $\|\boldsymbol{\Lambda}_{t,\text{new}} - \boldsymbol{\Lambda}_{t+1,\text{new}}\|_2$, then clearly, $\frac{\|\boldsymbol{\Sigma}_{t+1}-\boldsymbol{\Sigma}_t\|_2}{\|\boldsymbol{\Sigma}_t\|_2} \leq (3^{1/d} - 1)$ for all $t \in [t_j, t_j + d]$ and all j [2]. Since d is large, the upper bound is a small quantity, i.e. the covariance matrix changes slowly. For later time instants, we do not have any requirement (and so in particular $\boldsymbol{\Sigma}_t$ could still change slowly). Hence the above model includes "slow changing" and low-rank $\boldsymbol{\Sigma}_t$ as a special case.

2.2.2 Model on the set of missing entries or the outlier support set, $\mathcal{T}_t$

Our result requires that the set of missing entries (or the outlier support sets), $\mathcal{T}_t$, have *some* changes over time. We give one simple model for it below. One example that satisfies this model is a video application consisting of a foreground with one object of length s or less that can remain static for at most β frames at a time. When it moves, it moves *downwards* (or upwards, but always in one direction) by at least s/ρ pixels, and at most s/ρ_2 pixels. Once

[2] This follows because $\|\boldsymbol{\Sigma}_t\|_2 \geq \|\boldsymbol{\Lambda}_{t,\text{new}}\|_2 = \max_i (v_i)^{t-t_j} q_i \hat{\lambda}^-_{\text{train}}$ and $\|\boldsymbol{\Sigma}_{t+1} - \boldsymbol{\Sigma}_t\|_2 \leq \|\boldsymbol{\Lambda}_{t+1,\text{new}} - \boldsymbol{\Lambda}_{t,\text{new}}\|_2 \leq \max_i (v_i)^{t-t_j} q_i \hat{\lambda}^-_{\text{train}} (v_i - 1) \leq \max_i (v_i)^{t-t_j} q_i \hat{\lambda}^-_{\text{train}} \max_i (v_i - 1)$. Thus the ratio is bounded by $\max_i (v_i - 1) \leq (3/q_i)^{1/d} - 1 < (3^{1/d} - 1)$ since $q_i \geq 1$.

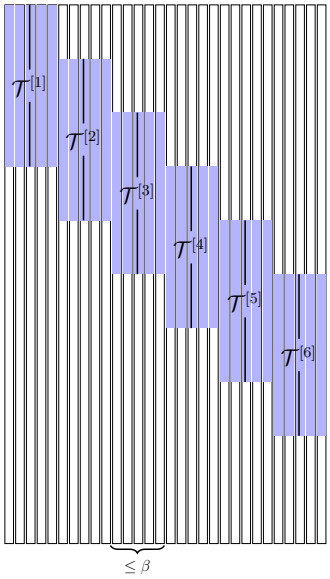

Figure 2.3 $\varrho = 3$ and $\beta = 5$ case

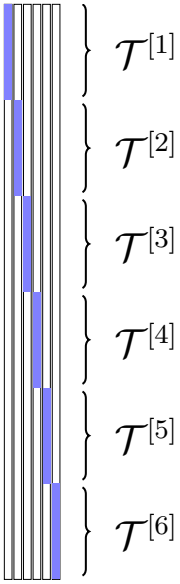

Figure 2.4 $\varrho = 1$ and $\beta = 1$ case

Figure 2.5: Examples of Model 2.2.3. (a) shows a 1D video object of length s that moves by at least $s/3$ pixels once every 5 frames. (b) shows the object moving by s at every frame. (b) is an example of the best case for our result - the case with smallest ρ, β ($\mathcal{T}_t$'s mutually disjoint)

it reaches the bottom of the scene, it disappears. The maximum motion is such that, if the object were to move at each frame, it still does not go from the top to the bottom of the scene in a time interval of length α, i.e. $\frac{s}{\rho_2}\alpha \leq n$. Anytime after it has disappeared another object could appear. We show this example in Fig. 2.5.

Model 2.2.3 (model on $\mathcal{T}_t$)**.** *Let t^k, with $t^k < t^{k+1}$, denote the times at which $\mathcal{T}_t$ changes and let $\mathcal{T}^{[k]}$ denote the distinct sets. For an integer α (we set its value in Theorem 2.2.7), assume the following.*

1. *Assume that $\mathcal{T}_t = \mathcal{T}^{[k]}$ for all times $t \in [t^k, t^{k+1})$ with $(t^{k+1} - t^k) \leq \beta$ and $|\mathcal{T}^{[k]}| \leq s$.*

2. *Let ρ be a positive integer so that for any k,*

$$\mathcal{T}^{[k]} \cap \mathcal{T}^{[k+\rho]} = \emptyset;$$

assume that

$$\rho^2 \beta \leq 0.01\alpha.$$

3. *For any k,*

$$\sum_{i=k+1}^{k+\alpha} \left| \mathcal{T}^{[i]} \setminus \mathcal{T}^{[i+1]} \right| \leq n$$

and for any $k < i \leq k + \alpha$,

$$(\mathcal{T}^{[k]} \setminus \mathcal{T}^{[k+1]}) \cap (\mathcal{T}^{[i]} \setminus \mathcal{T}^{[i+1]}) = \emptyset.$$

(One way to ensure $\sum_{i=k+1}^{k+\alpha} |\mathcal{T}^{[i]} \setminus \mathcal{T}^{[i+1]}| \leq n$ is to require that for all i, $|\mathcal{T}^{[i]} \setminus \mathcal{T}^{[i+1]}| \leq \frac{s}{\rho_2}$ with $\frac{s}{\rho_2}\alpha \leq n$.)

In this model, k takes values $1, 2, \ldots$; the largest value it can take is $t_{\max}$ (this will happen if $\mathcal{T}_t$ changes at every time).

Clearly the video moving object example satisfies the above model as long as $\rho^2\beta \leq 0.01\alpha$. [3] This becomes clearer from Fig. 2.5.

2.2.3 Denseness

In order to recover the $\boldsymbol{\ell}_t$'s from missing data or to separate them from the sparse outliers, the basis vectors for the subspace from which they are generated cannot be sparse. We quantify this using the incoherence condition from [1]. Let μ be the smallest real number such that

$$\max_i \|\boldsymbol{P}_{t_0}{}'\boldsymbol{I}_i\|_2^2 \leq \frac{\mu r_0}{n} \quad \text{and} \quad \max_i \|\boldsymbol{P}_{t_j,\text{new}}{}'\boldsymbol{I}_i\|_2^2 \leq \frac{\mu r_{j,\text{new}}}{n} \text{ for all } j \tag{2.7}$$

Recall from the notation section that $\boldsymbol{I}_i$ is the i^{th} column of the identity matrix (or i^{th} standard basis vector). We bound μr_0 and $\mu r_{j,\text{new}}$ in the theorem.

2.2.4 Main Result for Online Matrix Completion

Definition 2.2.4. *Recall that $r_{j,\text{new}} := \text{rank}(\boldsymbol{P}_{t_j,\text{new}})$ and $r_j := \text{rank}(\boldsymbol{P}_{t_j})$. Define $r_{\text{new}} := \max_j r_{j,\text{new}}$, and $r = r_0 + Jr_{\text{new}}$.*

[3] Let $\mathcal{T}_t$ be the support set of the object (set of pixels containing the object). The first condition holds since there is at most one object of size s or less and the object cannot remain static for more than β frames. Since it moves in one direction by at least s/ρ each time it moves, this means that definitely after it moves ρ times, the supports will be disjoint (second condition). The third condition holds because it moves in one direction and by at most s/ρ_2 with $\frac{s}{\rho_2}\alpha \leq n$ (so even if it were to move at each t, i.e. if $t_{k+1} = t_k + 1$ for all k, the third condition will hold). Also see Fig. 2.5.

Also define $\boldsymbol{a}_t := \boldsymbol{P}_t'\boldsymbol{\ell}_t$, *and for* $t \in [t_j, t_{j+1})$, $\boldsymbol{a}_{t,\text{new}} := \boldsymbol{P}_{t_j,\text{new}}'\boldsymbol{\ell}_t$. *Let*

$$\gamma := \max_t \|\boldsymbol{a}_t\|_\infty \quad \text{and} \quad \gamma_{\text{new}} := \max_j \max_{t\in[t_j,t_j+d]} \|\boldsymbol{a}_{t,\text{new}}\|_\infty$$

Notice that $\text{rank}(\boldsymbol{L}) = \text{rank}(\boldsymbol{P}_{t_{\max}}) \le r$. *Also,* $\|\boldsymbol{a}_t\|_2 \le \sqrt{r}\gamma$ *and for* $t \in [t_j, t_j + d]$, $\|\boldsymbol{a}_{t,\text{new}}\|_2 \le \sqrt{r_{\text{new}}}\gamma_{\text{new}}$.

The following theorem gives a correctness result for Algorithm 1 given and explained in Section 2.4. The algorithm has two parameters - α and K. The parameter α is the number of consecutive time instants that are used to obtain *an* estimate of the new subspace, and K is the total number of times the new subspace is estimated before we get an accurate enough estimate of it. The algorithm uses $\hat{\lambda}^-_{\text{train}}$ and $\hat{\boldsymbol{P}}_{t_{\text{train}}}$ defined in Definition 2.2.1 and $\boldsymbol{m}_t$ as inputs.

Theorem 2.2.5. *Consider Algorithm 1. Assume that* $\boldsymbol{m}_t$ *satisfies* (2.1). *Pick a* ζ *that satisfies*

$$\zeta \le \min\left\{\frac{10^{-4}}{r^2}, \frac{0.03\hat{\lambda}^-_{\text{train}}}{r^2\lambda^+}, \frac{1}{r^3\gamma^2}, \frac{\hat{\lambda}^-_{\text{train}}}{r^3\gamma^2}\right\}.$$

Suppose that the following hold.

1. $\text{dif}(\hat{\boldsymbol{P}}_{t_{\text{train}}}, \boldsymbol{P}_{t_{\text{train}}}) \le r_0\zeta$ *(notice from Model 2.2.2 that* $\boldsymbol{P}_{t_{\text{train}}} = \boldsymbol{P}_{t_0} = \boldsymbol{P}_1$*);*

2. *The algorithm parameters are set as:*
   $K = \left\lceil \frac{\log(0.16 r_{\text{new}}\zeta)}{\log(0.83)} \right\rceil$; *and* $\alpha = C(\log(6(K+1)J) + 11\log(n))$ *for a constant*
   $$C \ge C_{\text{add}} := 32 \cdot 100^2 \frac{\max\{16, 1.2(\sqrt{\zeta} + \sqrt{r_{\text{new}}}\gamma_{\text{new}})^4\}}{\left(r_{\text{new}}\zeta\hat{\lambda}^-_{\text{train}}\right)^2}; \tag{2.8}$$

3. *(Subspace change) Model 2.2.2 on* $\boldsymbol{\ell}_t$ *holds;*

4. *(Changes in the missing/corrupted sets* $\mathcal{T}_t$*) Model 2.2.3 on* $\mathcal{T}_t$ *holds or its generalization, Model 2.5.1 (given in Section 2.5), holds;*

5. *(Denseness and bound on* s, r_0, r_{new}*) the bounds in* (2.7) *hold with* $2s(r_0 + Jr_{\text{new}})\mu \le 0.09n$ *and* $2sr_{\text{new}}\mu \le 0.0004n$;

Then, with probability at least $1 - n^{-10}$, *at all times* t,

1. $\|\hat{\boldsymbol{\ell}}_t - \boldsymbol{\ell}_t\|_2 \le 1.2\left(\sqrt{\zeta} + \sqrt{r_{\text{new}}}\gamma_{\text{new}}\right)$

2. *the subspace error* $\mathrm{SE}_t := \|(\boldsymbol{I} - \hat{\boldsymbol{P}}_t \hat{\boldsymbol{P}}_t')\boldsymbol{P}_t\|_2$ *is bounded above by* $10^{-2}\sqrt{\zeta}$ *for* $t \in [t_j + d, t_{j+1})$.

Proof. The proof is given in Sections 2.6 and 2.7. As shown in Lemma 2.5.2, Model 2.2.3 is a special case of Model 2.5.1 (Model 2.5.1 is more general) on $\mathcal{T}_t$. Hence we prove the result only using Model 2.5.1. ⊠

Theorem 2.2.5 says that if an accurate estimate of the initial subspace is available; the two algorithm parameters are set appropriately; the $\boldsymbol{\ell}_t$'s are mutually independent over time and the low-dimensional subspace from which $\boldsymbol{\ell}_t$ is generated changes "slowly" enough, i.e. (a) the delay between change times is large enough ($d \geq (K+2)\alpha$) and (b) the eigenvalues along the newly added directions are small enough for d frames after a subspace change (so that (3b) holds); the set of missing entries at time t, $\mathcal{T}_t$, has enough changes; and the basis vectors that span the low-dimensional subspaces are dense enough; then, with high probability (w.h.p.), the error in estimating $\boldsymbol{\ell}_t$ will be small at all times t. Also, the error in estimating the low-dimensional subspace will be initially large when new directions are added, but will decay to a small constant times $\sqrt{\zeta}$ within a finite delay.

Consider the accurate initial subspace assumption. If the training data truly satisfies $\boldsymbol{m}_t = \boldsymbol{\ell}_t$ (without any noise or modeling error) and if we have at least r_0 linearly independent $\boldsymbol{\ell}_t$'s (if $\boldsymbol{\ell}_t$'s are continuous random vectors, this corresponds to needing $t_{\text{train}} \geq r_0$ almost surely), then the estimate of range$(\boldsymbol{P}_{t_{\text{train}}})$ obtained from training data will actually be exact, i.e. we will have $\mathrm{dif}(\hat{\boldsymbol{P}}_{t_{\text{train}}}, \boldsymbol{P}_{t_{\text{train}}}) = 0$. The theorem assumption that $\mathrm{dif}(\hat{\boldsymbol{P}}_{t_{\text{train}}}, \boldsymbol{P}_{t_{\text{train}}}) \leq r_0\zeta$ allows for the initial training data to be noisy or not exactly satisfying the model. If the training data is noisy, we need to know r_0 (in practice this is computed by thresholding to retain a certain percentage of largest eigenvalues). In this case we can let $\hat{\lambda}^-_{\text{train}}$ be the r_0-th eigenvalue of $\frac{1}{\alpha}\sum_{t=1}^{t_{\text{train}}} \boldsymbol{m}_t \boldsymbol{m}_t'$ and $\hat{\boldsymbol{P}}_{t_{\text{train}}}$ be the r_0 top eigenvectors.

The following corollary is also proved when we prove the above result.

Corollary 2.2.6. *The following conclusions also hold under the assumptions of Theorem 2.2.5 or 2.2.7 with probability at least* $1 - n^{-10}$

1. *The estimates of the subspace change times given by Algorithm 1 satisfy* $t_j \leq \hat{t}_j \leq t_j + 2\alpha$, *for* $j = 1, \ldots, J$;

2. *The estimates of the number of new directions are correct, i.e.* $\hat{r}_{j,\text{new},k} = r_{j,\text{new}}$ *for* $j = 1, \ldots, J$ *and* $k = 1, \ldots, K$;

3. *The recovery error satisfies:*

$$\|\hat{\boldsymbol{\ell}}_t - \boldsymbol{\ell}_t\|_2 \leq \begin{cases} 1.2\left(\sqrt{\zeta} + \sqrt{r_{\text{new}}}\gamma_{\text{new}}\right) & t \in \left[t_j, \hat{t}_j\right] \\ 1.2\left(1.84\sqrt{\zeta} + (0.83)^{k-1}\sqrt{r_{\text{new}}}\gamma_{\text{new}}\right) & t \in \left[\hat{t}_j + (k-1)\alpha, \hat{t}_j + k\alpha - 1\right], \\ & k = 1, 2, \ldots, K \\ 2.4\sqrt{\zeta} & t \in \left[\hat{t}_j + K\alpha, t_{j+1} - 1\right]; \end{cases}$$

4. *The subspace error satisfies,*

$$\text{SE}_t \leq \begin{cases} 1 & t \in \left[t_j, \hat{t}_j\right] \\ 10^{-2}\sqrt{\zeta} + 0.83^{k-1} & t \in \left[\hat{t}_j + (k-1)\alpha, \hat{t}_j + k\alpha - 1\right], \\ & k = 1, 2, \ldots, K \\ 10^{-2}\sqrt{\zeta} & t \in \left[\hat{t}_j + K\alpha, t_{j+1} - 1\right]. \end{cases}$$

2.2.5 Main Result for Online Robust PCA

Recall that in this case we assume that the observations $\boldsymbol{m}_t$ satisfy $\boldsymbol{m}_t = \boldsymbol{\ell}_t + \boldsymbol{x}_t$ with the support of $\boldsymbol{x}_t$, denoted $\mathcal{T}_t$, not known. We have the following result for Algorithm 2 given and explained in Section 2.4. This requires two extra assumptions beyond what the previous result needed. For the matrix completion problem, the set of missing entries is known, while in the robust PCA setting, the support set, $\mathcal{T}_t$, of the sparse outliers, $\boldsymbol{x}_t$, must be determined. We recover this using an ℓ_1 minimization step followed by thresholding. To do this correctly, we need a lower bound on the absolute values of the nonzero entries of $\boldsymbol{x}_t$. Moreover, Algorithm 2 has two extra parameters - ξ, which is the bound on the two norm of the noise seen by the ℓ_1 minimization step, and ω, which is the threshold used to recover the support of $\boldsymbol{x}_t$. These need to be set appropriately.

Theorem 2.2.7. *Consider Algorithm 2. Assume that $\boldsymbol{m}_t$ satisfies (2.2) and assume everything else in Theorem 2.2.5. Also assume*

1. *The two extra algorithm parameters are set as: $\xi = \sqrt{r_{\text{new}}}\gamma_{\text{new}} + (\sqrt{r} + \sqrt{r_{\text{new}}})\sqrt{\zeta}$ and $\omega = 7\xi$*

2. *We have $x_{\min} := \min_t \min_{i:(\boldsymbol{x}_t)_i \neq 0} |(\boldsymbol{x}_t)_i| > 14\xi$*

Then with probability at least $1 - n^{-10}$,

1. *all conclusions of Theorem 2.2.5 and Corollary 2.2.6 hold;*

2. *the support set $\mathcal{T}_t$ is exactly recovered, i.e. $\hat{\mathcal{T}}_t = \mathcal{T}_t$ for all t;*

3. *$\|\boldsymbol{x}_t - \hat{\boldsymbol{x}}_t\|_2 = \|\boldsymbol{\ell}_t - \hat{\boldsymbol{\ell}}_t\|_2$ and $\|\boldsymbol{\ell}_t - \hat{\boldsymbol{\ell}}_t\|_2$ satisfies the bounds given in Theorem 2.2.5 and Corollary 2.2.6.*

The second assumption above can be interpreted as either a lower bound on $x_{\min}$, or as an upper bound on $\sqrt{r_{\text{new}}}\gamma_{\text{new}}$ in terms of $x_{\min}$. This latter interpretation is another "slow subspace change" condition. For the $\boldsymbol{x}_t$'s, this result shows that their support is exactly recovered w.h.p. and its nonzero entries are accurately recovered.

2.2.6 Simple Generalizations

Consider the subspace change model, Model 2.2.2. For simplicity we put a slow increase model on the eigenvalues along the new directions for the entire period $[t_j, t_{j+1})$. However, as explained below the model, the proof of our result does not actually use this slow increase model. It only uses (3b), i.e. $\hat{\lambda}^-_{\text{train}} \leq \lambda^-_{\text{new}} \leq \lambda^+_{\text{new}} \leq 3\hat{\lambda}^-_{\text{train}}$. Recall that λ^-_{new} and λ^+_{new} are the minimum and maximum eigenvalues along the new directions for the first d frames after a subspace change. Thus, in the interval $[t_j + d + 1, t_{j+1})$ our proof actually does not need any constraint on $\boldsymbol{\Lambda}_{t,\text{new}}$.

With a minor modification to our proof, we can prove our result with an even weaker condition. We need (3b) to hold with λ^-_{new} being the minimum of the minimum eigenvalues of any α-frame *average* covariance matrix along the new directions over the period $[t_j, t_j + d]$, i.e.

with $\lambda^{-}_{\text{new}} = \min_j \min_{\tau \in [t_j, t_j+d-\alpha]} \lambda_{\min}(\frac{1}{\alpha}\sum_{t=\tau}^{\tau+\alpha-1} \mathbf{\Lambda}_{t,\text{new}})$. For video analytics, this translates to requiring that, after a subspace change, *enough (but not necessarily all)* background frames have 'detectable' energy along the new directions, so that the minimum eigenvalue of the average covariance is above a threshold.

Secondly, we should point out that there is a trade off between the bound on $q_i {v_i}^d$, and consequently on λ^{+}_{new}, in Model 2.2.2 and the bound on $\rho^2\beta$ assumed in Model 2.2.3. Allowing a larger value of $q_i {v_i}^d$ (faster subspace change) will require a tighter bound on $\rho^2\beta$ which corresponds to requiring more changes to $\mathcal{T}_t$. We chose the bounds $q_i (v_i)^d \leq 3$ and $\rho^2\beta \leq .01\alpha$ for simplicity of computations. There are many other pairs that would also work. The above trade-off can be seen from the proof of Lemma 2.6.14. The proof uses Model 2.5.1 of which Model 2.2.3 is a special case. For video analytics, this means that if the background subspace changes are faster, then we also need the foreground objects to be moving more so we can 'see' enough of the background behind them.

Thirdly, in Model 2.2.2 we let $\boldsymbol{P}_t \mathbf{\Lambda}_t \boldsymbol{P}_t{}'$ be an EVD of $\mathbf{\Sigma}_t$. This automatically implies that $\mathbf{\Lambda}_t$ is diagonal. But our proof only uses the fact that $\mathbf{\Lambda}_t$ is block diagonal with blocks $\mathbf{\Lambda}_{t,*}$ and $\mathbf{\Lambda}_{t,\text{new}}$. If we relax this and we let $\boldsymbol{P}_t \mathbf{\Lambda}_t \boldsymbol{P}_t{}'$ be *a* decomposition of $\mathbf{\Sigma}_t$ where $\mathbf{\Lambda}_t$ is block diagonal as above, then our model allows the variance along *any* direction from range$(\boldsymbol{P}_{t_j-1})$ to become zero for any period of time and/or become nonzero again later. Thus, in the special case of (2.6) we can actually allow $\boldsymbol{P}_t = [(\boldsymbol{P}_{t-1}\boldsymbol{R}_t \setminus \boldsymbol{P}_{t,\text{del}}) \;\; \boldsymbol{P}_{t,\text{new}}]$, where $\boldsymbol{R}_t$ is an $r_{j-1} \times r_{j-1}$ rotation matrix and $\boldsymbol{P}_{t,\text{del}}$ contains the columns of $\boldsymbol{P}_{t-1}\boldsymbol{R}_t$ for which the variance is zero. This will be a special case of this generalization if $[\boldsymbol{P}_{t_1-1} \; \boldsymbol{P}_{t_1,\text{new}} \; \boldsymbol{P}_{t_2,\text{new}} \; \ldots \; \boldsymbol{P}_{t_J,\text{new}}]$ is a basis matrix.

Lastly, the first condition of the theorem requires that we have accurate initial subspace knowledge. As explained below the theorem, this means that we can allow noisy training data. Moreover, notice that if we let $t_1 = t_{\text{train}} + 1$, then new background directions can enter the subspace at the same time as the first foreground object. Said another way, all we need is an accurate enough estimate of all but r_{new} directions of the initial subspace, and an assumption of small eigenvalues for sometime (d frames) along the directions for which we do not have an accurate enough estimate (or do not have an estimate).

Now consider the denseness assumption. Define the (un)denseness coefficient as follows.

Definition 2.2.8. *For a basis matrix* $\boldsymbol{P}$*, define* $\kappa_s(\boldsymbol{P}) := \max_{|\mathcal{T}|\leq s} \|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{P}\|_2$.

Notice that left hand side in (2.7) is $[\kappa_1(\boldsymbol{P})]^2$. Using the triangle inequality, it is easy to show that $\kappa_s(\boldsymbol{P}) \leq \sqrt{s}\kappa_1(\boldsymbol{P})$ [11]. Therefore, using the fact that for a basis matrix $[\boldsymbol{P}_1 \ \boldsymbol{P}_2]$, $(\kappa_s([\boldsymbol{P}_1 \ \boldsymbol{P}_2]))^2 \leq (\kappa_s(\boldsymbol{P}_1))^2 + (\kappa_s(\boldsymbol{P}_2))^2$ (see proof of the first statement of Lemma 2.C.2 in Appendix 2.C), the denseness assumptions of Theorem 2.2.7 imply that

$$\kappa_{s,*} := \kappa_{2s}(\boldsymbol{P}_{t_J}) \leq 0.3 \quad \text{and} \quad \kappa_{s,\text{new}} := \max_j \kappa_{2s}(\boldsymbol{P}_{t_j,\text{new}}) \leq 0.02. \tag{2.9}$$

The proof of Theorem 2.2.7 only uses (2.9) for the denseness assumption.

The reason for defining κ_s as above is the following lemma from [11].

Lemma 2.2.9 ([11])**.** *For a basis matrix* $\boldsymbol{P}$*,* $\delta_s(\boldsymbol{I} - \boldsymbol{P}\boldsymbol{P}') = (\kappa_s(\boldsymbol{P}))^2$.

2.3 Discussion

2.3.1 Discussion of the assumptions used

In the previous section, we provide two related results, one for online matrix completion (MC) and the second for online robust PCA (RPCA). The result for online RPCA can also be interpreted as a result for online sparse matrix recovery in (potentially) large but structured noise $\boldsymbol{\ell}_t$. Notice that our result does not require an upper bound on λ^+ (the maximum eigenvalue of $\text{Cov}(\boldsymbol{\ell}_t)$ at any time) or on γ (the bound on the maximum magnitude of any entry of $\boldsymbol{P}_t'\boldsymbol{\ell}_t$ for any time t). Both these parameters are only used to select ζ, which in turn governs the value of K and α and hence governs the required delay between subspace change times.

Our results require accurate initial subspace knowledge. As explained earlier, for video analytics, this corresponds to requiring an initial short sequence of background-only video frames whose subspace can be estimated via SVD (followed by using a singular value threshold to retain a certain number of top left singular vectors). Alternatively if an initial short sequence of the video data satisfies the assumptions required by a batch method such as PCP (for RPCA) and NNM (for MC), that can be used to estimate the low-rank part, followed by SVD to get

the column subspace. For online MC, another alternative is to use the initialization techniques of GROUSE [7] or PETRELS [8] or to use the adaptive MC idea of [10].

In Model 2.2.2, we are placing a slow increase assumption on the eigenvalues along the new directions, $\boldsymbol{P}_{t_j,\text{new}}$, for the interval $[t_j, t_{j+1})$. Thus after t_{j+1}, the eigenvalues along $\boldsymbol{P}_{t_j,\text{new}}$ can increase gradually or suddenly to any large value up to λ^+. In fact as explained above, our proof needs the slow increase to hold only for the first d time instants after t_j, so, in fact, at any time after $t_j + d$, the eigenvalues along $\boldsymbol{P}_{t_j,\text{new}}$ could increase to a large value.

Model 2.2.3 on $\mathcal{T}_t$ is a practical model for moving foreground objects in video. We should point out that this model is one special case of the general set of conditions we need (Model 2.5.1). Some other special cases of it are discussed in Section 2.9.

The model on $\mathcal{T}_t$ (Model 2.2.3) and the denseness condition of the theorem constrain s and $s, r_0, r_{\text{new}}, J$ respectively. Model 2.2.3 requires $s \leq \rho_2 n/\alpha$ for a constant ρ_2. Using the expression for α, it is easy to see that as long as $J \in \mathcal{O}(n)$, we have $\alpha \in \mathcal{O}(\log n)$ and so Model 2.2.3 needs $s \in \mathcal{O}(\frac{n}{\log n})$. With $s \in \mathcal{O}(\frac{n}{\log n})$, the denseness condition will hold if $r_0 \in \mathcal{O}(\log n)$, $J \in \mathcal{O}(\log n)$ and r_{new} is a constant. This is one set of sufficient conditions that we allow on the rank-sparsity product.

2.3.2 Comparison with the results for PCP and NNM

Let $\boldsymbol{L} := [\boldsymbol{\ell}_1, \boldsymbol{\ell}_2, \ldots, \boldsymbol{\ell}_{t_{\max}}]$ and $\boldsymbol{S} := [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{t_{\max}}]$. Let $r_{\text{mat}} := \text{rank}(\boldsymbol{L})$. Clearly $r_{\text{mat}} \leq r_0 + J r_{\text{new}}$ and the bound is tight. Let $s_{\text{mat}} := t_{\max} s$ be a bound on the total number of missing entries of $\boldsymbol{L}$ or on the support size of the outliers' matrix $\boldsymbol{S}$. In terms of r_{mat} and s_{mat}, what we need is $r_{\text{mat}} \in \mathcal{O}(\log n)$ and $s_{\text{mat}} \in \mathcal{O}(\frac{n t_{\max}}{\log n})$. This is stronger than what the PCP result from [1] or the result for NNM from [5] need (e.g., the PCP result from [1] allows $r_{\text{mat}} \in \mathcal{O}\left(\frac{n}{(\log n)^2}\right)$ while allowing $s_{\text{mat}} \in \mathcal{O}(n t_{\max})$), but is similar to what the PCP results from [2, 3] need.

Other disadvantages of our result are as follows. (1) Our result needs accurate initial subspace knowledge and slow subspace change of $\boldsymbol{\ell}_t$. As explained earlier and in [11, Fig. 6], both of these are often practically valid for video analytics applications. Moreover, we also need the $\boldsymbol{\ell}_t$'s to be zero mean and mutually independent over time. Zero mean is achieved

by letting $\boldsymbol{\ell}_t$ be the background image at time t with an empirical 'mean background image', computed using the training data, subtracted out. The independence assumption then models independent background variations around a common mean. As we explain in Section 2.9, this can be easily relaxed and we can get a result very similar to the current one under a first order autoregressive model on the $\boldsymbol{\ell}_t$'s. (2) Moreover, Algorithms 1 and 2 need multiple algorithm parameters to be appropriately set. The PCP or NNM results need this for none [1, 5] or at most one [2, 3] algorithm parameter. (3) Thirdly, our result for online RPCA also needs a lower bound on $x_{\min}$ while the PCP results do not need this. (4) Moreover, even with this, we can only guarantee accurate recovery of $\boldsymbol{\ell}_t$, while PCP or NNM guarantee exact recovery.

The advantages of our work are (1) that we analyze an online algorithm (ReProCS) that is faster and needs less storage compared with PCP or NNM. It needs to store only a few $n \times \alpha$ or $n \times r_{\text{mat}}$ matrices, thus the storage complexity is $\mathcal{O}(n \log n)$ while that for PCP or NNM is $\mathcal{O}(nt_{\max})$. In general $t_{\max}$ can be much larger than $\log n$. (2) Moreover, we do not need any assumption on the right singular vectors of $\boldsymbol{L}$ while all results for PCP or NNM do. (3) Most importantly, our results allow highly correlated changes of the set of missing entries (or outliers). From the assumption on $\mathcal{T}_t$, it is easy to see that we allow the number of missing entries (or outliers) per row of $\boldsymbol{L}$ to be $\mathcal{O}(t_{\max})$ as long as the sets follow Model 2.2.3[4]. The PCP results from [2, 3] need this number to be $\mathcal{O}(\frac{t_{\max}}{r_{\text{mat}}})$ which is stronger. The PCP result from [1] or the NNM result [5] need an even stronger condition - they need the set $(\cup_{t=1}^{t_{\max}} \mathcal{T}_t)$ to be generated uniformly at random.

2.3.3 Other results for online RPCA and online MC

Our online RPCA result improves upon the online RPCA results from our earlier work [11] for two reasons. First, the result of [11] was a *partial result* because it required a denseness assumption on $(\boldsymbol{I} - \boldsymbol{P}_{t_j,\text{new}} \boldsymbol{P}_{t_j,\text{new}}{}')\hat{\boldsymbol{P}}_t$ and $(\boldsymbol{I} - \hat{\boldsymbol{P}}_{t,*}\hat{\boldsymbol{P}}_{t,*}{}' - \hat{\boldsymbol{P}}_{t,\text{new}}\hat{\boldsymbol{P}}_{t,\text{new}}{}')\boldsymbol{P}_{t_j,\text{new}}$. Here $\hat{\boldsymbol{P}}_{t,*}$ and $\hat{\boldsymbol{P}}_{t,\text{new}}$ are estimates computed by Algorithm 2. Thus, the result depended on intermediate algorithm estimates satisfying certain properties. In this work, we remove this requirement

[4] In a period of length α, the set $\mathcal{T}_t$ can occupy index i for at most $\rho\beta$ time instants, and this pattern is allowed to repeat every α time instants. So an index can be in the support for a total of $\rho\beta\frac{t_{\max}}{\alpha}$ time instants and the model assumes $\rho\beta \le \frac{0.01\alpha}{\rho}$ for a constant ρ.

and instead provide a *complete correctness result*. The extra assumption that we need is Model 2.2.3 on $\mathcal{T}_t$ (or its generalization given in Model 2.5.1 later). Secondly, we provide a correctness result for a ReProCS-based algorithm that detects subspace change automatically and also estimates the rank of the new subspace automatically. The algorithm studied in [11] required knowing t_j and $r_{j,\text{new}}$ exactly for each j. Algorithms 1 and 2 in this work only require upper bounds on r_{new}, γ_{new} and J (these are needed to set the algorithm parameters - α and K for Algorithm 1, and also ξ and ω for Algorithm 2) and a small enough ζ (need bounds on r, λ^+ and γ to set this). A third minor advantage is that we also provide an algorithm and a result for online MC.

The proof of our results adapts the overall framework developed in [11]. The two important additions are: (a) Model 2.5.1 and Lemma 2.5.3 for it, and the way it is used in the proof of Lemma 2.6.23; and (b) the detection lemma (Lemma 2.6.17), the no false detection lemma (Lemma 2.6.16) and the p-PCA lemma (Lemma 2.6.18) and the lemmas used to prove these. (a) allows us to get a complete correctness result; (b) allows us to analyze an algorithm that does not use knowledge of t_j or $r_{j,\text{new}}$.

In [20], Feng et. al. propose a method for online RPCA and prove a partial result for their algorithm. The approach is to reformulate the PCP program and use this reformulation to develop a recursive algorithm that converges asymptotically to the solution of PCP as long as the basis estimate $\hat{\boldsymbol{P}}_t$ is full rank at each time t. Since this result assumes something about the algorithm estimates, it is also only a *partial* result.

Another recent work that uses knowledge of the initial subspace estimate and performs recovery in a piecewise batch fashion is modified-PCP [21]. However, like PCP, the result for modified PCP also needs uniformly randomly generated support sets. Its advantage is that its assumption on the rank-sparsity product is weaker than that of PCP, and hence weaker than that needed by this work. A detailed simulation comparison between modified-PCP, ReProCS and PCP demonstrating both these things is available in [21, Fig. 6].

Some other recent works that also study the online MC problem (defined differently from how we define it) include [6], Grassmanian Rank-One Update Subspace Estimation (GROUSE) [7] and Parallel Subspace Estimation and Tracking by Recursive Least Squares From Partial

Observations (PETRELS) [8]. We discuss the connection with [6] in Section 2.4. GROUSE is a first order stochastic gradient method. It uses rank-one updates to track the underlying subspace on the Grassmannian manifold. A result for its convergence to the local minimum of the cost function it optimizes is obtained in [9]. PETRELS is a second order stochastic gradient method. As explained in [8], in PETRELS, the low-dimensional subspace is tracked by minimizing a geometrically discounted sum of projection residuals on the observed entries at each time index. If missing entries are required then they can be reconstructed via least squares estimation. The subspace is updated recursively so that it is not necessary to retain historical data indefinitely. If the underlying subspace is fixed and the data stream is fully observed, then it is shown that the PETRELS estimate converges to the true subspace. In general, it always converges to the stationary point of the cost function it optimizes [8]. The advantage of PETRELS and GROUSE is that they do not need initial subspace knowledge. For our algorithms, when the initial subspace knowledge is not available or initial complete and outlier-free data is not available, we can also use the PETRELS or GROUSE ideas for initialization.

2.4 Automatic ReProCS Algorithms for Online MC and Online RPCA and Why They Work

In this section, we first introduce the automatic ReProCS based algorithm for online MC and explain why it works (this also provides the key idea why the proof of our main result would go through). Next, we do the same thing for the online RPCA algorithm. In the last two subsections (Sections 2.4.3 and 2.4.4), we explain the key insight used by our proof and give the proof outline.

2.4.1 Automatic ReProCS for Online MC (Algorithm 1)

The model on $\boldsymbol{m}_t$ from (2.1) is a special case of that from (2.2) with $\boldsymbol{x}_t = -\boldsymbol{I}_{\mathcal{T}_t}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{\ell}_t$ and with the support of $\boldsymbol{x}_t$, $\mathcal{T}_t$ known [1]. Thus, we can use a simplification of the ReProCS idea for online RPCA [11] to also solve the online MC problem.

Algorithm 1 proceeds as follows. Let $\hat{\boldsymbol{P}}_{t-1}$ denote the basis matrix for the estimate of the subspace where $\boldsymbol{\ell}_{t-1}$ lies. If it is an accurate estimate, because of "slow subspace change", projecting the measurement $\boldsymbol{m}_t = \boldsymbol{x}_t + \boldsymbol{\ell}_t$ onto its orthogonal complement will nullify most of $\boldsymbol{\ell}_t$. Specifically, we compute $\boldsymbol{y}_t := \boldsymbol{\Phi}_t \boldsymbol{m}_t$ where $\boldsymbol{\Phi}_t := \boldsymbol{I} - \hat{\boldsymbol{P}}_{t-1}\hat{\boldsymbol{P}}_{t-1}{}'$. Thus, $\boldsymbol{y}_t$ can be rewritten as

$$\boldsymbol{y}_t = \boldsymbol{\Phi}_t \boldsymbol{x}_t + \boldsymbol{b}_t \text{ where } \boldsymbol{b}_t := \boldsymbol{\Phi}_t \boldsymbol{\ell}_t$$

and it can be argued that $\|\boldsymbol{b}_t\|_2$ is small. Since the support of $\boldsymbol{x}_t$, $\mathcal{T}_t$, is known, we can simply recover its nonzero entries by least squares (LS) estimation, i.e. we get $\hat{\boldsymbol{x}}_t = \boldsymbol{I}_{\mathcal{T}_t}(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}^{\dagger}\boldsymbol{y}_t$ and then get an estimate of $\boldsymbol{\ell}_t$ as $\hat{\boldsymbol{\ell}}_t = \boldsymbol{m}_t - \hat{\boldsymbol{x}}_t$. The above approach of recovering $\boldsymbol{\ell}_t$ is equivalent to that used by Brand in [6]; there they recover $\boldsymbol{\ell}_t$ as an LS estimate of $\hat{\boldsymbol{P}}\hat{\boldsymbol{P}}'\boldsymbol{\ell}_t \approx \boldsymbol{\ell}_t$.

Let $\boldsymbol{e}_t := \boldsymbol{\ell}_t - \hat{\boldsymbol{\ell}}_t$. With the above, it is easy to see that

$$\boldsymbol{e}_t = \boldsymbol{I}_{\mathcal{T}_t}(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}^{\dagger}\boldsymbol{b}_t = \boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{\Phi}\boldsymbol{\ell}_t.$$

Using the denseness assumption, it can be argued that the RIC of $\boldsymbol{\Phi}_t$ will be small (see Lemma 2.2.9). Under the theorem's assumptions, and conditioned on accurate recovery so far, we can bound it by 0.14. Thus, $\|(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}^{-1}\|_2 \leq 1/(1-0.14) < 1.2$ and so $\|\boldsymbol{e}_t\|_2 \leq 1.2\|\boldsymbol{b}_t\|_2$, i.e. it is small too (see Lemma 2.6.15).

Projection-PCA (p-PCA). The next step is to use a modification of standard PCA called projection-PCA (p-PCA), to update the subspace estimate. The reason we need p-PCA is this. Let $\sum_t$ denote a sum over an α length time interval. In our problem, the error, $\boldsymbol{e}_t$, in the observation/estimate of $\boldsymbol{\ell}_t$, $\hat{\boldsymbol{\ell}}_t$, is correlated with $\boldsymbol{\ell}_t$. Because of this, the dominant terms in the perturbation seen by standard PCA, $\frac{1}{\alpha}\sum_t \hat{\boldsymbol{\ell}}_t\hat{\boldsymbol{\ell}}_t{}' - \frac{1}{\alpha}\sum_t \boldsymbol{\ell}_t\boldsymbol{\ell}_t{}'$, are $\frac{1}{\alpha}\sum_t \boldsymbol{\ell}_t\boldsymbol{e}_t{}'$ and its transpose[5]. Thus, when the condition number of $\mathrm{Cov}(\boldsymbol{\ell}_t)$ is large, it becomes difficult to argue that the perturbation will be small compared to the smallest eigenvalue of $\mathrm{Cov}(\boldsymbol{\ell}_t)$. With a large perturbation, either the $\sin\theta$ theorem [22] (that bounds the subspace error between the eigenvectors of the true and estimated sample covariance matrices) cannot be applied or it gives a useless bound.

[5]When $\boldsymbol{\ell}_t$ and $\boldsymbol{e}_t$ are uncorrelated and one of them is zero mean, it can be argued by law of large numbers that, whp, these two terms will be close to zero and $\frac{1}{\alpha}\sum_t \boldsymbol{e}_t\boldsymbol{e}_t{}'$ will be the dominant term.

Our proposed approach, projection-PCA (p-PCA) addresses the above issue as follows. At $t = t_j$, let $\boldsymbol{P}_* := \boldsymbol{P}_{t_{j-1}}$, $\boldsymbol{P}_{\text{new}} := \boldsymbol{P}_{t_j,\text{new}}$, and suppose that the subspace range($\boldsymbol{P}_*$) has been accurately recovered, i.e. we have $\hat{\boldsymbol{P}}_*$ so that $\text{dif}(\hat{\boldsymbol{P}}_*, \boldsymbol{P}_*) \ll 1$. Then at a time at or after $t_j+\alpha$ if we project the α previous $\hat{\boldsymbol{\ell}}_t$'s perpendicular to $\hat{\boldsymbol{P}}_*$, we will considerably reduce the perturbation seen by the PCA step. We detect subspace change by checking if the maximum singular value of the matrix formed by these projected $\hat{\boldsymbol{\ell}}_t$'s is above a threshold. Denote the time at which change is detected by $\hat{t}_j$. After $\hat{t}_j$ we use SVD on K different sets of α frames of the projected $\hat{\boldsymbol{\ell}}_t$'s to get improved estimates of the new subspace range($\boldsymbol{P}_{\text{new}}$) in each iteration. To be precise, we get the k-th estimate, $\hat{\boldsymbol{P}}_{\text{new},k}$, as the left singular vectors of $(\boldsymbol{I} - \hat{\boldsymbol{P}}_*\hat{\boldsymbol{P}}_*')[\hat{\boldsymbol{\ell}}_{\hat{t}_j+(k-1)\alpha+1}, \dots, \hat{\boldsymbol{\ell}}_{\hat{t}_j+k\alpha}]$ with singular values above a threshold. After each p-PCA step, we update $\hat{\boldsymbol{P}}_t$ as $\hat{\boldsymbol{P}}_t = [\hat{\boldsymbol{P}}_* \ \hat{\boldsymbol{P}}_{\text{new},k}]$. Finally at time $t = \hat{t}_j + K\alpha$, we update $\hat{\boldsymbol{P}}_*$ as $[\hat{\boldsymbol{P}}_* \ \hat{\boldsymbol{P}}_{\text{new},K}]$.

In the subspace update step, Algorithm 1 toggles between the "detect" phase and the "ppca" phase. It starts in the "detect" phase. When a subspace change is detected, i.e. at $t = \hat{t}_j$ it enters the "ppca" phase. After K iterations of p-PCA, i.e. at $t = \hat{t}_j + K\alpha + 1$, the new subspace has been accurately estimated and this time it enters the "detect" phase again.

Why p-PCA works. The reason p-PCA works is as follows. Before the first p-PCA step, i.e. for $t \in [t_j, \hat{t}_j + \alpha)$, $\hat{\boldsymbol{P}}_t = \hat{\boldsymbol{P}}_*$ and thus the noise seen by the projected sparse recovery step, $\boldsymbol{b}_t = \boldsymbol{\Phi}\boldsymbol{\ell}_t = (\boldsymbol{I} - \hat{\boldsymbol{P}}_*\hat{\boldsymbol{P}}_*')\boldsymbol{\ell}_t$, will be the largest. Hence the error $\boldsymbol{e}_t$ will also be the largest for the $\hat{\boldsymbol{\ell}}_t$'s used for the first p-PCA step. However because of the projection perpendicular to $\hat{\boldsymbol{P}}_*$ and slow subspace change, even this error is not too large. Because of this and because $\boldsymbol{e}_t$ is sparse and supported on $\mathcal{T}_t$ and $\mathcal{T}_t$ follows Model 2.2.3, we can argue that $\hat{\boldsymbol{P}}_{\text{new},1}$ is a good estimate, i.e. $\text{dif}([\hat{\boldsymbol{P}}_* \ \hat{\boldsymbol{P}}_{\text{new},1}], \boldsymbol{P}_{\text{new}}) \le 0.2 < 1$. After the first p-PCA step, $\hat{\boldsymbol{P}}_t = [\hat{\boldsymbol{P}}_* \ \hat{\boldsymbol{P}}_{\text{new},1}]$ and this will reduce $\boldsymbol{b}_t$ and hence $\boldsymbol{e}_t$ for the $\hat{\boldsymbol{\ell}}_t$'s in the next α frames. This and the sparseness of $\boldsymbol{e}_t$, in turn, will mean that the perturbation seen by the second p-PCA step will be smaller and so $\hat{\boldsymbol{P}}_{\text{new},2}$ will be a more accurate estimate of range($\boldsymbol{P}_{\text{new}}$) than $\hat{\boldsymbol{P}}_{\text{new},1}$. This is done K times with K chosen so that $\text{dif}([\hat{\boldsymbol{P}}_* \ \hat{\boldsymbol{P}}_{\text{new},K}], \boldsymbol{P}_{\text{new}}) \le r_{\text{new}}\zeta$. By the theorem assumptions, and because we can show $t_j \le \hat{t}_j < t_j + 2\alpha$ (we explain this below), it is clear that $t_{j+1} > \hat{t}_j + K\alpha$. Thus, the new subspace added at t_j is accurately estimated before the next change time t_{j+1}.

Why $\hat{t}_j$ are correctly detected. As explained above, we detect subspace changes by comparing the eigenvalues of $(\boldsymbol{I}-\hat{\boldsymbol{P}}_*\hat{\boldsymbol{P}}_*{}')\frac{1}{\alpha}\sum_t \hat{\boldsymbol{\ell}}_t\hat{\boldsymbol{\ell}}_t{}'(\boldsymbol{I}-\hat{\boldsymbol{P}}_*\hat{\boldsymbol{P}}_*{}')$ to a chosen threshold at every $t = u\alpha$ for $u = 1, 2, \ldots, \lfloor \frac{t_{\max}}{\alpha} \rfloor$ when the algorithm is in the "detect" phase. In order to correctly detect $\hat{t}_j$, the algorithm first must not falsely detect new directions when none are present and it must detect subspace change within a short delay after it has occurred. The former will occur because conditioned on accurate recovery of the current subspace, $(\boldsymbol{I}-\hat{\boldsymbol{P}}_*\hat{\boldsymbol{P}}_*{}')\frac{1}{\alpha}\sum_t \hat{\boldsymbol{\ell}}_t\hat{\boldsymbol{\ell}}_t{}'(\boldsymbol{I}-\hat{\boldsymbol{P}}_*\hat{\boldsymbol{P}}_*{}')$ will have very small eigenvalues when no new directions are present. If the recovery were exact and no new directions present, this matrix would be zero. In our case, the recovery is only accurate and so we show that all eigenvalues of this matrix will be below the chosen threshold (see Lemma 2.6.16). Next consider detection of the subspace change after it has occurred. When $u = u_j := \left\lceil \frac{t_j}{\alpha} \right\rceil$, i.e. when t_j is in the interval $\big((u-1)\alpha+1, u\alpha\big]$, not all of the $\boldsymbol{\ell}_t$'s in this interval will contain new directions. Thus, depending on where in the interval t_j lies, the algorithm may or may not detect the subspace change. However, in the *next* interval, $[u_j\alpha+1, (u_j+1)\alpha]$, all of the $\boldsymbol{\ell}_t$'s will contain new directions, and we can prove that the subspace change will be detected w.h.p. (see Lemma 2.6.17). Thus, w.h.p., either $\hat{t}_j = u_j\alpha$, or $\hat{t}_j = (u_j+1)\alpha$. Thus, we will be able to show that $t_j \le \hat{t}_j \le t_j + 2\alpha$ w.h.p..

A visual description of Algorithm 1 is shown in Fig. 2.6. This figure uses Definition 2.6.4.

Algorithm 1 ReProCS for Online MC

Parameters: α, K, *Inputs*: $\hat{\boldsymbol{P}}_{t_{\text{train}}}$, $\hat{\lambda}^{-}_{\text{train}}$, $\boldsymbol{m}_t$ for each t, *Output*: $\hat{\boldsymbol{\ell}}_t$, $\hat{\boldsymbol{P}}_t$, $\hat{t}_{\hat{\jmath}}$, $\hat{r}_{\hat{\jmath},\text{new},k}$
Let thresh $= \frac{\hat{\lambda}^{-}_{\text{train}}}{2}$ (this is the eigenvalue threshold that will be used to detect subspace change).
Set $\hat{\boldsymbol{P}}_{t,*} \leftarrow \hat{\boldsymbol{P}}_{t_{\text{train}}}$, $\hat{\boldsymbol{P}}_{t,\text{new}} \leftarrow [.]$, $\hat{\jmath} \leftarrow 0$, phase $\leftarrow$ detect.
For every $t > t_{\text{train}}$, do the following:

- Compute $\boldsymbol{y}_t \leftarrow \boldsymbol{\Phi}_t \boldsymbol{m}_t$ where $\boldsymbol{\Phi}_t \leftarrow \boldsymbol{I} - \hat{\boldsymbol{P}}_{t-1}\hat{\boldsymbol{P}}_{t-1}{}'$
- Estimate $\boldsymbol{\ell}_t$: $\hat{\boldsymbol{\ell}}_t \leftarrow \boldsymbol{m}_t - \boldsymbol{I}_{\mathcal{T}_t}((\boldsymbol{\Phi}_t)_{\mathcal{T}_t})^{\dagger}\boldsymbol{y}_t$
- If $t \mod \alpha \neq 0$ then $\hat{\boldsymbol{P}}_{t,*} \leftarrow \hat{\boldsymbol{P}}_{t-1,*}$, $\hat{\boldsymbol{P}}_{t,\text{new}} \leftarrow \hat{\boldsymbol{P}}_{t-1,\text{new}}$, $\hat{\boldsymbol{P}}_t \leftarrow [\hat{\boldsymbol{P}}_{t,*}\ \hat{\boldsymbol{P}}_{t,\text{new}}]$
- If $t \mod \alpha = 0$ then *detection or projection PCA*
  If phase = detect then
  1. Set $u = \frac{t}{\alpha}$ and compute $\boldsymbol{\mathcal{D}}_u = (\boldsymbol{I} - \hat{\boldsymbol{P}}_{u\alpha-1,*}\hat{\boldsymbol{P}}_{u\alpha-1,*}{}')[\hat{\boldsymbol{\ell}}_{(u-1)\alpha+1}, \dots \hat{\boldsymbol{\ell}}_{u\alpha}]$
  2. $\hat{\boldsymbol{P}}_{t,*} \leftarrow \hat{\boldsymbol{P}}_{t-1,*}$, $\hat{\boldsymbol{P}}_{t,\text{new}} \leftarrow \hat{\boldsymbol{P}}_{t-1,\text{new}}$, $\hat{\boldsymbol{P}}_t \leftarrow [\hat{\boldsymbol{P}}_{t,*}\ \hat{\boldsymbol{P}}_{t,\text{new}}]$
  3. If $\lambda_{\max}(\frac{1}{\alpha}\boldsymbol{\mathcal{D}}_u\boldsymbol{\mathcal{D}}_u{}') \geq$ thresh then
     phase $\leftarrow$ ppca, $\hat{\jmath} \leftarrow \hat{\jmath} + 1$, $k \leftarrow 0$, $\hat{t}_{\hat{\jmath}} = t$

  Else (phase = ppca) then
  1. Set $u = \frac{t}{\alpha}$ and compute $\boldsymbol{\mathcal{D}}_u = (\boldsymbol{I} - \hat{\boldsymbol{P}}_{u\alpha-1,*}\hat{\boldsymbol{P}}_{u\alpha-1,*}{}')[\hat{\boldsymbol{\ell}}_{(u-1)\alpha+1}, \dots \hat{\boldsymbol{\ell}}_{u\alpha}]$
  2. $\hat{\boldsymbol{P}}_{t,\text{new}} \leftarrow$ eigenvectors $\left(\frac{1}{\alpha}\boldsymbol{\mathcal{D}}_u\boldsymbol{\mathcal{D}}_u{}', \text{thresh}\right)$, $\hat{\boldsymbol{P}}_{t,*} \leftarrow \hat{\boldsymbol{P}}_{t-1,*}$, $\hat{\boldsymbol{P}}_t \leftarrow [\hat{\boldsymbol{P}}_{t,*}\ \hat{\boldsymbol{P}}_{t,\text{new}}]$
  3. $k \leftarrow k + 1$, set $\hat{r}_{j,\text{new},k} = \text{rank}(\hat{\boldsymbol{P}}_{t,\text{new}})$
  4. If $k = K$, then
     phase $\leftarrow$ detect, $\hat{\boldsymbol{P}}_{t,*} \leftarrow \hat{\boldsymbol{P}}_t$, $\hat{\boldsymbol{P}}_{t,\text{new}} \leftarrow [.]$

eigenvectors$(\boldsymbol{\mathcal{M}}, \text{thresh})$ returns a basis matrix for the span of all eigenvectors whose eigenvalue is above thresh.

2.4.2 Automatic ReProCS for online RPCA (Algorithm 2)

For online RPCA the only difference is that the support for $\boldsymbol{x}_t$, $\mathcal{T}_t$, is not known. Hence we first recover $\boldsymbol{x}_t$ by ℓ_1 minimization (or any other sparse recovery method) and then estimate its support by thresholding. The rest of the steps remain the same as above.

2.4.3 Key Insight for the Proof

The argument given while explaining why p-PCA works in Section 2.4 can be formalized to show that, w.h.p., a subspace change is detected only after a change has occurred and

Algorithm 2 ReProCS for Online RPCA

Parameters: α, K, ξ, ω, *Inputs*: $\hat{\boldsymbol{P}}_{t_{\text{train}}}$, $\hat{\lambda}^{-}_{\text{train}}$, $\boldsymbol{m}_t$ for each t, *Output*: $\hat{\boldsymbol{\ell}}_t$, $\hat{\boldsymbol{P}}_t$, $\hat{t}_{\hat{\jmath}}$
Let thresh $= \frac{\hat{\lambda}^{-}_{\text{train}}}{2}$. Set $\hat{\boldsymbol{P}}_{t,*} \leftarrow \hat{\boldsymbol{P}}_{t_{\text{train}}}$, $\hat{\boldsymbol{P}}_{t,\text{new}} \leftarrow [.]$, $\hat{\jmath} \leftarrow 0$, phase $\leftarrow$ detect.
For every $t > t_{\text{train}}$, do the following:

- Estimate $\mathcal{T}_t$ (the support of the outlier vector $\boldsymbol{x}_t$) and $\boldsymbol{x}_t$.
  1. compute $\boldsymbol{y}_t \leftarrow \boldsymbol{\Phi}_t \boldsymbol{m}_t$ where $\boldsymbol{\Phi}_t \leftarrow \boldsymbol{I} - \hat{\boldsymbol{P}}_{t-1}\hat{\boldsymbol{P}}_{t-1}{}'$
  2. solve $\min_{\boldsymbol{x}} \|\boldsymbol{x}\|_1$ *s.t.* $\|\boldsymbol{y}_t - \boldsymbol{\Phi}_t \boldsymbol{x}\|_2 \le \xi$ and let $\hat{\boldsymbol{x}}_{t,\text{cs}}$ denote its solution
  3. compute $\hat{\mathcal{T}}_t = \{i : \ |(\hat{\boldsymbol{x}}_{t,\text{cs}})_i| > \omega\}$
  4. LS estimate of $\boldsymbol{x}_t$: compute $\hat{\boldsymbol{x}}_t = \boldsymbol{I}_{\hat{\mathcal{T}}_t}((\boldsymbol{\Phi}_t)_{\hat{\mathcal{T}}_t})^{\dagger}\boldsymbol{y}_t$
- Use all steps of Algorithm 1 with $\mathcal{T}_t \leftarrow \hat{\mathcal{T}}_t$.

within 2α frames of the change; and that the subspace recovery error, SE_t, will decay roughly exponentially with each p-PCA iteration and become small enough after K iterations. To do this we will use the $\sin\theta$ theorem [22] (Lemma 2.6.20) followed by the matrix Hoeffding inequality [23] (Lemmas 2.7.5, 2.7.6)) to get high probability bounds on each of the terms in the subspace error bound obtained by the $\sin\theta$ theorem.

While applying the matrix Hoeffding inequality, we need to use the following key insight about the structure of $\mathbb{E}[\frac{1}{\alpha}\sum_t (\boldsymbol{I} - \hat{\boldsymbol{P}}_*\hat{\boldsymbol{P}}_*{}')\boldsymbol{\ell}_t \boldsymbol{e}_t{}']$. This matrix is the dominant term in the perturbation seen by the k-th p-PCA step. Here $\mathbb{E}[.]$ denotes expectation conditioned on accurate subspace recovery so far and $\sum_t$ denotes the sum over $t \in [\hat{t}_j + (k-1)\alpha + 1, \hat{t}_j + k\alpha]$. The model on $\mathcal{T}_t$ and the fact that $\boldsymbol{e}_t$ is supported on $\mathcal{T}_t$ can be used to show that this matrix can be written as the product of a full matrix and a block-banded matrix: for example when $\rho = 1$, the block-banded matrix will be block-diagonal, when $\rho = 2$, it will be block-tridiagonal, and so on. Also, $\mathbb{E}[\frac{1}{\alpha}\sum_t \boldsymbol{e}_t \boldsymbol{e}_t{}']$ will be a block banded matrix. The 2-norm of a block banded matrix is bounded by the maximum norm of any block times the number of bands in it and hence is much smaller than that of a general full matrix.

The lemma that exploits the structure of a block-banded matrix generated due to the model on $\mathcal{T}_t$ is Lemma 2.5.3 given in Sec 2.5. This lemma is used to bound $\mathbb{E}[\frac{1}{\alpha}\sum_t (\boldsymbol{I} - \hat{\boldsymbol{P}}_*\hat{\boldsymbol{P}}_*{}')\boldsymbol{\ell}_t \boldsymbol{e}_t{}']$ and $\mathbb{E}[\frac{1}{\alpha}\sum_t \boldsymbol{e}_t \boldsymbol{e}_t{}']$ in the proof of Lemma 2.6.23 in Section 2.7.

2.4.4 Proof Outline

We will only prove Theorem 2.2.7. Theorem 2.2.5 follows as a corollary of Theorem 2.2.7 because of the following reasons. (1) Algorithm 1 does not compute $\hat{\boldsymbol{x}}_t$ or its support $\hat{\mathcal{T}}_t$. For the matrix completion problem, $\mathcal{T}_t$ is given. Thus it does not use the parameters ξ (which is the noise bound in the ℓ_1 minimization step) and ω (which is the support estimation threshold). (2) The bound on $x_{\min}$ and the values of the parameters ξ and ω are only used in the proof of Lemma 2.6.15 to show exact support recovery, i.e $\hat{\mathcal{T}}_t = \mathcal{T}_t$. Since for matrix completion $\mathcal{T}_t$ is given, Theorem 2.2.5 does not need need the lower bound on $x_{\min}$.

The proof of Theorem 2.2.7 is given in Sections 2.6 and 2.7. Before this, in the next section (Section 2.5) we give the most general model on changes in the missing/outlier entries' set $\mathcal{T}_t$, Model 2.5.1, and we show that Model 2.2.3 is a special case of this model. Next, we give a key lemma for sums of sparse matrices supported on rows and columns indexed by $\mathcal{T}_t$ satisfying this model (Lemma 2.5.3).

Section 2.6 begins with defining various quantities needed for the proof. Next, we state the main lemmas used to prove the theorem, followed by the theorem's proof. There is a main lemma associated with each of the three main tasks of the algorithm: 1) accurately recovering $\boldsymbol{x}_t$ and hence $\boldsymbol{\ell}_t$ at each time t (Lemma 2.6.15), 2) detecting (subspace change) when and only when the subspace has changed, i.e. new directions have been added to the subspace (Lemmas 2.6.17 and 2.6.16), and 3) successfully estimating the dimension of the new subspace and updating its estimate by p-PCA (Lemma 2.6.18). To maintain the flow of the argument, we defer the proofs of these lemmas to the end of the section or to the appendix.

The proofs of Lemmas 2.6.21, 2.6.22, and 2.6.23 that are used together to prove Lemmas 2.6.17, 2.6.16 and 2.6.18 are rather long and are given in section 2.7. The proof of Lemma 2.6.23 uses Lemma 2.5.3 from Section 2.5.

2.5 Most General Model on Changes in $\mathcal{T}_t$ and a Key Lemma

2.5.1 Most General Model on Changes in $\mathcal{T}_t$

Here we give our most general model on how $\mathcal{T}_t$ (the set of missing entries or the support set of $\boldsymbol{x}_t$) can change. What we need to prevent is $\mathcal{T}_t$ occupying the same indices for too many time instants in a given interval. If $\mathcal{T}_t$ does not change 'enough' in a time interval of length α, we will be unable to see enough entries of a given index of $\boldsymbol{\ell}_t$ in order to be able to accurately fill in the missing ones.

The following model quantifies 'enough' for our purposes. The number of time instants for which an index is part of $\mathcal{T}_t$ is determined both by how often this set changes, and by how much it moves when it changes. The latter is parameterized by ρ which controls how much the set moves when it changes. For example $\rho = 1$ would require that distinct sets be disjoint, and $\rho = 2$ would mean that at least half of the set is displaced whenever it changes. The parameter $h^+ \in (0, 1)$ represents the maximum fraction of time for which the set remains in a given area in a time interval of length α. The smaller h^+, the more frequently the set will need to change in order to satisfy the model. Our result requires a bound on the product $\rho^2 h^+$.

Model 2.5.1. *Let ρ be a positive integer. Split $[1, t_{\max}]$ into intervals of length α. Use $\mathcal{J}_u := [(u-1)\alpha+1, u\alpha]$ to denote the u-th interval. For a given interval, $\mathcal{J}_u$, let $\mathcal{T}_{(i),u}$ for $i = 1, \ldots, l_u$ be mutually disjoint subsets of $\{1, \ldots, n\}$ and let $\mathcal{J}_{(i),u}, i = 1, 2, \ldots, l_u$ be a partition[6] of the interval $\mathcal{J}_u$ so that*

$$\text{for all } t \in \mathcal{J}_{(i),u},\ \mathcal{T}_t \subseteq \mathcal{T}_{(i),u} \cup \mathcal{T}_{(i+1),u} \cup \cdots \cup \mathcal{T}_{(i+\rho-1),u}\,. \tag{2.10}$$

Define

$$h_u\left(\alpha; \{\mathcal{T}_{(i),u}\}_{i=1,\ldots,l_u}, \{\mathcal{J}_{(i),u}\}_{i=1,\ldots,l_u}\right) := \max_{i=1,2,\ldots l_u} \left|\mathcal{J}_{(i),u}\right| \tag{2.11}$$

and define $h_u^*(\alpha)$ which takes the minimum over all choices of $\mathcal{T}_{(i),u}$ and over all choices of the

[6] *i.e. the $\mathcal{J}_{(i),u}$'s are mutually disjoint intervals and their union equals $\mathcal{J}_u$*

partition $\mathcal{J}_{(i),u}$.

$$h_u^*(\alpha) := \min_{\substack{\text{all choices of mutually disjoint } \mathcal{T}_{(i),u}, i=1,2,\dots l_u \\ \text{and all choices of mutually disjoint } \mathcal{J}_{(i),u}, i=1,2,\dots l_u \\ \text{so that } \cup_{i=1}^{l_u} \mathcal{J}_{(i),u} = \mathcal{J}_u \text{ and (2.10) holds}}} h_u\left(\alpha; \{\mathcal{T}_{(i),u}\}_{i=1,\dots,l_u}, \{\mathcal{J}_{(i),u}\}_{i=1,\dots,l_u}\right) \tag{2.12}$$

Assume that $|\mathcal{T}_t| \leq s$ *and that for all* $u = 1, \dots, \lceil \frac{t_{\max}}{\alpha} \rceil$,

$$h_u^*(\alpha) \leq h^+\alpha \quad \textit{with } h^+ \leq \frac{0.01}{\rho^2}.$$

In the above model, $h_u^*(\alpha)$ provides a bound on how long $\mathcal{T}_t$ remains in a given "area", $\mathcal{T}_{(i),u} \cup \mathcal{T}_{(i+1),u} \cup \cdots \cup \mathcal{T}_{(i+\rho-1),u}$ during the interval $\mathcal{J}_u$, for the best allocation of $\mathcal{T}_t$'s to a given "area" and the best choice of the "areas."

Notice that (2.10) can always be trivially satisfied by choosing $l_u = 1$, $\mathcal{T}_{(1),u} = \{1, \dots, n\}$ and $\mathcal{J}_{(1),u} = \mathcal{J}_u$, but this will give $h_u(\alpha; .) = \alpha$ and hence is not a good choice. This is why we take a minimum over all choices.

Lemma 2.5.2. *Model 2.2.3 is a special case of Model 2.5.1 above with* $h^+ = \frac{\beta}{\alpha}$.

The proof is in Appendix 2.A.

Some other special cases of the above model are discussed in Section 2.9.

2.5.2 A Key Lemma that uses Model 2.5.1

Lemma 2.5.3. *Let* $s_t = |\mathcal{T}_t|$. *Consider a sequence of* $s_t \times s_t$ *symmetric positive-semidefinite matrices* $\boldsymbol{A}_t$ *such that* $\|\boldsymbol{A}_t\|_2 \leq \sigma^+$ *for all* t. *Assume that the* $\mathcal{T}_t$ *obey Model 2.5.1. Let* $\boldsymbol{M} = \sum_{t \in \mathcal{J}_u} \boldsymbol{I}_{\mathcal{T}_t} \boldsymbol{A}_t \boldsymbol{I}_{\mathcal{T}_t}'$ *be an* $n \times n$ *matrix (*$\boldsymbol{I}$ *is an* $n \times n$ *identity matrix). Then*

$$\|\boldsymbol{M}\|_2 \leq \rho^2 h^+ \alpha \sigma^+ \leq 0.01 \sigma^+ \alpha$$

Proof. We will first prove the lemma for the special case when $\rho = 2$. After this, we will show how to generalize the proof when $\rho > 2$. For a given u, let $\mathcal{T}_{(i),u}$, $i = 1, 2, \dots l_u$, and correspondingly $\mathcal{J}_{(i),u}$ denote the best choices, i.e. the choices that attain the minimum values in the definition of $h_u^*(\alpha)$.

In the rest of the proof, we remove the subscript u from l_u and from $\mathcal{T}_{(i),u}$'s for ease of notation. For simplicity of notation, we will let $\mathcal{T}_{(l+1),u} = \emptyset$.

For times $t \in \mathcal{J}_{(i),u}$, define $\boldsymbol{A}_{t,\text{full}}$ to be $\boldsymbol{A}_t$ with rows and columns of zeros appropriately inserted so that

$$\boldsymbol{I}_{\mathcal{T}_t} \boldsymbol{A}_t \boldsymbol{I}_{\mathcal{T}_t}{}' = \boldsymbol{I}_{\mathcal{T}_{(i)} \cup \mathcal{T}_{(i+1)}} \boldsymbol{A}_{t,\text{full}} \boldsymbol{I}_{\mathcal{T}_{(i)} \cup \mathcal{T}_{(i+1)}}{}'. \tag{2.13}$$

Such an $\boldsymbol{A}_{t,\text{full}}$ exists because $\mathcal{T}_t \subseteq \mathcal{T}_{(i)} \cup \mathcal{T}_{(i+1)}$ for any $t \in \mathcal{J}_{(i),u}$. Notice that

$$\|\boldsymbol{A}_{t,\text{full}}\|_2 = \|\boldsymbol{A}_t\|_2, \tag{2.14}$$

because $\boldsymbol{A}_{t,\text{full}}$ is permutation similar to

$$\begin{bmatrix} \boldsymbol{A}_t & 0 \\ 0 & 0 \end{bmatrix}.$$

Since $\mathcal{T}_{(i)}$ and $\mathcal{T}_{(i+1)}$ are disjoint, we can, after permutation similarity, correspondingly partition $\boldsymbol{A}_{t,\text{full}}$ as

$$\begin{bmatrix} \boldsymbol{A}_{t,\text{full}}^{(0,0)} & \boldsymbol{A}_{t,\text{full}}^{(0,1)} \\ \boldsymbol{A}_{t,\text{full}}^{(1,0)} & \boldsymbol{A}_{t,\text{full}}^{(1,1)} \end{bmatrix}$$

for all $t \in \mathcal{J}_{(i),u}$.

Notice that because $\boldsymbol{A}_t$ is symmetric, $\boldsymbol{A}_{t,\text{full}}^{(1,0)} = \left(\boldsymbol{A}_{t,\text{full}}^{(0,1)}\right)'$. Then,

$$
\begin{aligned}
\boldsymbol{M} &= \sum_{t\in\mathcal{J}_u} \boldsymbol{I}_{\mathcal{T}_t} \boldsymbol{A}_t \boldsymbol{I}_{\mathcal{T}_t}{}' \\
&= \sum_{i=1}^{l} \sum_{t\in\mathcal{J}_{(i),u}} \boldsymbol{I}_{\mathcal{T}_{(i)}\cup\mathcal{T}_{(i+1)}} \boldsymbol{A}_{t,\text{full}} \boldsymbol{I}_{\mathcal{T}_{(i)}\cup\mathcal{T}_{(i+1)}}{}' \qquad \text{by (2.13)} \\
&= \sum_{i=1}^{l} \sum_{t\in\mathcal{J}_{(i),u}} [\boldsymbol{I}_{\mathcal{T}_{(i)}} \ \boldsymbol{I}_{\mathcal{T}_{(i+1)}}] \begin{bmatrix} \boldsymbol{A}_{t,\text{full}}^{(0,0)} & \boldsymbol{A}_{t,\text{full}}^{(0,1)} \\ \boldsymbol{A}_{t,\text{full}}^{(1,0)} & \boldsymbol{A}_{t,\text{full}}^{(1,1)} \end{bmatrix} \begin{bmatrix} \boldsymbol{I}_{\mathcal{T}_{(i)}}{}' \\ \boldsymbol{I}_{\mathcal{T}_{(i+1)}}{}' \end{bmatrix} \\
&= \sum_{i=1}^{l} \sum_{t\in\mathcal{J}_{(i),u}} \left[\boldsymbol{I}_{\mathcal{T}_{(i)}} \boldsymbol{A}_{t,\text{full}}^{(0,0)} \boldsymbol{I}_{\mathcal{T}_{(i)}}{}' + \boldsymbol{I}_{\mathcal{T}_{(i)}} \boldsymbol{A}_{t,\text{full}}^{(0,1)} \boldsymbol{I}_{\mathcal{T}_{(i+1)}}{}' + \boldsymbol{I}_{\mathcal{T}_{(i+1)}} \boldsymbol{A}_{t,\text{full}}^{(1,0)} \boldsymbol{I}_{\mathcal{T}_{(i)}}{}' + \boldsymbol{I}_{\mathcal{T}_{(i+1)}} \boldsymbol{A}_{t,\text{full}}^{(1,1)} \boldsymbol{I}_{\mathcal{T}_{(i+1)}}{}'\right] \\
&= \boldsymbol{I}_{\mathcal{T}_{(1)}} \left(\sum_{t\in\mathcal{J}_{(1),u}} \boldsymbol{A}_{t,\text{full}}^{(0,0)} \right) \boldsymbol{I}_{\mathcal{T}_{(1)}}{}' + \sum_{i=2}^{l} \left[\boldsymbol{I}_{\mathcal{T}_{(i)}} \left(\sum_{t\in\mathcal{J}_{(i-1),u}} \boldsymbol{A}_{t,\text{full}}^{(1,1)} + \sum_{t\in\mathcal{J}_{(i),u}} \boldsymbol{A}_{t,\text{full}}^{(0,0)} \right) \boldsymbol{I}_{\mathcal{T}_{(i)}}{}' \right] \\
&\quad + \boldsymbol{I}_{\mathcal{T}_{(l)}} \left(\sum_{t\in\mathcal{J}_{(l),u}} \boldsymbol{A}_{t,\text{full}}^{(1,1)} \right) \boldsymbol{I}_{\mathcal{T}_{(l)}}{}' \\
&\quad + \sum_{i=1}^{l-1} \left[\boldsymbol{I}_{\mathcal{T}_{(i)}} \left(\sum_{t\in\mathcal{J}_{(i),u}} \boldsymbol{A}_{t,\text{full}}^{(0,1)} \right) \boldsymbol{I}_{\mathcal{T}_{(i+1)}}{}' + \boldsymbol{I}_{\mathcal{T}_{(i+1)}} \left(\sum_{t\in\mathcal{J}_{(i),u}} \boldsymbol{A}_{t,\text{full}}^{(1,0)} \right) \boldsymbol{I}_{\mathcal{T}_{(i)}}{}' \right]
\end{aligned}
$$

Because $\mathcal{T}_{(i)}$ and $\mathcal{T}_{(k)}$ are disjoint for $i \neq k$, $\boldsymbol{M}$ has a block tridiagonal structure (by a permutation similarity if necessary):

$$
\begin{bmatrix} \boldsymbol{B}_{(1)} & \boldsymbol{C}_{(1)} & 0 & 0 \\ \boldsymbol{C}_{(1)}{}' & \boldsymbol{B}_{(2)} & \ddots & 0 \\ 0 & \ddots & \ddots & \boldsymbol{C}_{(l-1)} \\ 0 & 0 & \boldsymbol{C}_{(l-1)}{}' & \boldsymbol{B}_{(l)} \end{bmatrix} \tag{2.15}
$$

where $\boldsymbol{B}_{(1)} = \sum_{t\in\mathcal{J}_{(1),u}} \boldsymbol{A}_{t,\text{full}}^{(0,0)}$, $\boldsymbol{B}_{(l)} = \sum_{t\in\mathcal{J}_{(l),u}} \boldsymbol{A}_{t,\text{full}}^{(1,1)}$,

$$
\boldsymbol{B}_{(i)} = \sum_{t\in\mathcal{J}_{(i-1),u}} \boldsymbol{A}_{t,\text{full}}^{(1,1)} + \sum_{t\in\mathcal{J}_{(i),u}} \boldsymbol{A}_{t,\text{full}}^{(0,0)} \quad \text{for } i = 2, 3, \ldots, l \tag{2.16}
$$

and

$$
\boldsymbol{C}_{(i)} = \sum_{t\in\mathcal{J}_{(i),u}} \boldsymbol{A}_{t,\text{full}}^{(0,1)} \quad \text{for } i = 1, 2, \ldots, l-1. \tag{2.17}
$$

Now we proceed to bound $\|\boldsymbol{M}\|_2$.

$$\|\boldsymbol{M}\|_2 = \left\| \begin{array}{cccc} \boldsymbol{B}_{(1)} & \boldsymbol{C}_{(1)} & 0 & 0 \\ \boldsymbol{C}_{(1)}{}' & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & \boldsymbol{C}_{(l-1)} \\ 0 & 0 & \boldsymbol{C}_{(l-1)}{}' & \boldsymbol{B}_{(l)} \end{array} \right\|_2$$

$$\leq \left\| \begin{array}{cccc} \boldsymbol{B}_{(1)} & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \boldsymbol{B}_{(l)} \end{array} \right\|_2 + \left\| \begin{array}{cccc} 0 & \boldsymbol{C}_{(1)} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \boldsymbol{C}_{(l-1)} \\ 0 & 0 & 0 & 0 \end{array} \right\|_2 + \left\| \begin{array}{cccc} 0 & 0 & 0 & 0 \\ \boldsymbol{C}_{(1)}{}' & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 \\ 0 & 0 & \boldsymbol{C}_{(l-1)}{}' & 0 \end{array} \right\|_2 .$$

Call the middle matrix $\boldsymbol{C}$, and observe that $\boldsymbol{C}\boldsymbol{C}'$ is block diagonal with blocks $\boldsymbol{C}_{(i)}\boldsymbol{C}_{(i)}{}'$. So $\|\boldsymbol{C}\|_2 = \max_i \|\boldsymbol{C}_{(i)}\|_2$. Therefore,

$$\begin{aligned} \|\boldsymbol{M}\|_2 &\leq \max_i \|\boldsymbol{B}_{(i)}\|_2 + 2\max_i \|\boldsymbol{C}_{(i)}\|_2 \\ &\leq \max_i \left\| \sum_{t\in\mathcal{J}_{(i-1),u}} \boldsymbol{A}^{(1,1)}_{t,\text{full}} + \sum_{t\in\mathcal{J}_{(i),u}} \boldsymbol{A}^{(0,0)}_{t,\text{full}} \right\|_2 + 2\max_i \left\| \sum_{t\in\mathcal{J}_{(i),u}} \boldsymbol{A}^{(0,1)}_{t,\text{full}} \right\|_2 \qquad \text{by (2.16) and (2.17)} \\ &\leq \max_i \left(\sum_{t\in\mathcal{J}_{(i-1),u}} \left\|\boldsymbol{A}_t\right\|_2 + \sum_{t\in\mathcal{J}_{(i),u}} \left\|\boldsymbol{A}_t\right\|_2 \right) + 2\max_i \sum_{t\in\mathcal{J}_{(i),u}} \left\|\boldsymbol{A}_t\right\|_2 \qquad \text{by (2.14)} \\ &\leq (\sigma^+ h^*_u(\alpha) + \sigma^+ h^*_u(\alpha)) + 2\sigma^+ h^*_u(\alpha) \leq 4\sigma^+ h^+ \alpha \end{aligned}$$

The third row used the fact that $\|\boldsymbol{A}^{(\cdot,\cdot)}_{t,\text{full}}\|_2 \leq \|\boldsymbol{A}_{t,\text{full}}\|_2 = \|\boldsymbol{A}_t\|_2$ for any sub-matrix of $\boldsymbol{A}_{t,\text{full}}$.

This finishes the proof for the $\rho = 2$ case. For this case, notice that there are 3 bands in (2.15) - the diagonal band and one band on each side of the diagonal one. When $\rho = 3$, everything will follow analogously to the above; instead of 3 bands, there will be 5 bands in the definition of $\boldsymbol{M}$ and we will be able to bound its norm by

$$\begin{aligned}\|\boldsymbol{M}\|_2 \leq & \max_i \left(\sum_{t\in\mathcal{J}_{(i-2),u}} \left\|\boldsymbol{A}_t\right\|_2 + \sum_{t\in\mathcal{J}_{(i-1),u}} \left\|\boldsymbol{A}_t\right\|_2 + \sum_{t\in\mathcal{J}_{(i),u}} \left\|\boldsymbol{A}_t\right\|_2 \right) \\ & + 2\max_i \left(\sum_{t\in\mathcal{J}_{(i-1),u}} \left\|\boldsymbol{A}_t\right\|_2 + \sum_{t\in\mathcal{J}_{(i),u}} \left\|\boldsymbol{A}_t\right\|_2 \right) \\ & + 2\max_i \sum_{t\in\mathcal{J}_{(i),u}} \left\|\boldsymbol{A}_t\right\|_2 \\ \leq & 3\sigma^+ h_u^*(\alpha) + 2(2\sigma^+ h_u^*(\alpha) + \sigma^+ h_u^*(\alpha)) \leq 9\sigma^+ h^+ \alpha\end{aligned}$$

Proceeding this way, for a general ρ, there will be $1+2(\rho-1) = 2\rho-1$ bands. Any term in the central band will contain a summation of $\left\|\boldsymbol{A}_t\right\|_2$ over ρ sub-intervals $\mathcal{J}_{(i),u}$; any term in the first band away from the diagonal will contain this summation over $(\rho-1)$ sub-intervals; any term in the second band away from the diagonal will contain this summation over $(\rho-2)$ sub-intervals; and so on. Thus, we will be summing the quantity $\sigma^+ h^+ \alpha$ a total of $(\rho + 2\sum_{i=1}^{\rho-1} i) = \rho^2$ times and so we will get $\|\boldsymbol{M}\|_2 \leq \rho^2 \sigma^+ h^+ \alpha$. ⊠

2.6 Proof of Theorem 2.2.7 and Theorem 2.2.5

As explained in Section 2.4.4, we will only prove Theorem 2.2.7. Theorem 2.2.5 follows as an easy corollary.

2.6.1 Definitions

Definition 2.6.1. *Define $\boldsymbol{e}_t$ to be the error made in estimating $\boldsymbol{x}_t$ and $\boldsymbol{\ell}_t$. That is*

$$\boldsymbol{e}_t := \hat{\boldsymbol{x}}_t - \boldsymbol{x}_t = \boldsymbol{\ell}_t - \hat{\boldsymbol{\ell}}_t$$

Definition 2.6.2. *Define the interval*

$$\mathcal{J}_u := [(u-1)\alpha + 1, u\alpha].$$

Also define u_j to be the u such that $t_j \in \mathcal{J}_u$. That is

$$u_j := \left\lceil \frac{t_j}{\alpha} \right\rceil.$$

For the purposes of describing events before the first subspace change, let $u_0 := 0$. *Also define*

$$\hat{u}_j := \frac{\hat{t}_j}{\alpha}.$$

Notice from the algorithm that this will be an integer, because detection only occurs when $t \bmod \alpha = 0$.

We will show that, under appropriate conditioning, w.h.p., $\hat{u}_j = u_j$ or $\hat{u}_j = u_j + 1$.

Definition 2.6.3. *Define*

$$\boldsymbol{P}_{(j)} := \boldsymbol{P}_{t_j} \text{ for } j = 0, 1, \ldots, J$$

$$\boldsymbol{P}_{(j),*} := \boldsymbol{P}_{(j-1)} = \boldsymbol{P}_{t_{j-1}} \text{ and } \boldsymbol{P}_{(j),\text{new}} := \boldsymbol{P}_{t_j,\text{new}} \text{ for } j = 1, \ldots, J$$

$$\boldsymbol{a}_{t,*} := \boldsymbol{P}_{(j),*}{}'\boldsymbol{\ell}_t \text{ and } \boldsymbol{a}_{t,\text{new}} := \boldsymbol{P}_{(j),\text{new}}{}'\boldsymbol{\ell}_t \text{ for } t \in [t_j, t_{j+1})$$

Thus, for $t \in [t_j, t_{j+1})$, $\boldsymbol{\ell}_t$ can be written as

$$\boldsymbol{\ell}_t = [\boldsymbol{P}_{(j),*}\ \boldsymbol{P}_{(j),\text{new}}] \begin{bmatrix} \boldsymbol{a}_{t,*} \\ \boldsymbol{a}_{t,\text{new}} \end{bmatrix} = \boldsymbol{P}_{(j),*}\boldsymbol{a}_{t,*} + \boldsymbol{P}_{(j),\text{new}}\boldsymbol{a}_{t,\text{new}}$$

and $\text{Cov}(\boldsymbol{\ell}_t) = \boldsymbol{\Sigma}_t$ can be rewritten as

$$\boldsymbol{\Sigma}_t = \begin{bmatrix} \boldsymbol{P}_{(j),*}\ \boldsymbol{P}_{(j),\text{new}} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Lambda}_{t,*} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Lambda}_{t,\text{new}} \end{bmatrix} \begin{bmatrix} \boldsymbol{P}_{(j),*}{}' \\ \boldsymbol{P}_{(j),\text{new}}{}' \end{bmatrix}$$

Definition 2.6.4. *For* $j = 1, 2, \ldots, J$ *and* $k = 1, 2, \ldots, K$ *define*

1. $\hat{\boldsymbol{P}}_{(1),*} := \hat{\boldsymbol{P}}_{t_{\text{train}}}$ *(the initial estimate) and* $\hat{\boldsymbol{P}}_{(j),*} := \hat{\boldsymbol{P}}_{\hat{t}_{j-1}+K\alpha}$. *If all subspace changes are correctly detected, this is the final estimate of* $\boldsymbol{P}_{(j),*} = \boldsymbol{P}_{(j-1)}$.

2. $\hat{\boldsymbol{P}}_{(j),\text{new},0} := [.]$ *and* $\hat{\boldsymbol{P}}_{(j),\text{new},k} := \hat{\boldsymbol{P}}_{\hat{t}_j+k\alpha,\text{new}}$. *This is the* k^{th} *estimate of* $\boldsymbol{P}_{(j),\text{new}}$ *(again, conditioned on correct change time detection).*

Notice from the algorithm that

1. $\hat{\boldsymbol{P}}_{t,*} = \hat{\boldsymbol{P}}_{(j),*}$ for all $t \in [\hat{t}_{j-1} + K\alpha, \hat{t}_j + K\alpha - 1]$

2. $\hat{\boldsymbol{P}}_{t,\text{new}} = \hat{\boldsymbol{P}}_{(j),\text{new},k}$ for all $t \in \mathcal{J}_{\hat{u}_j+(k+1)}$

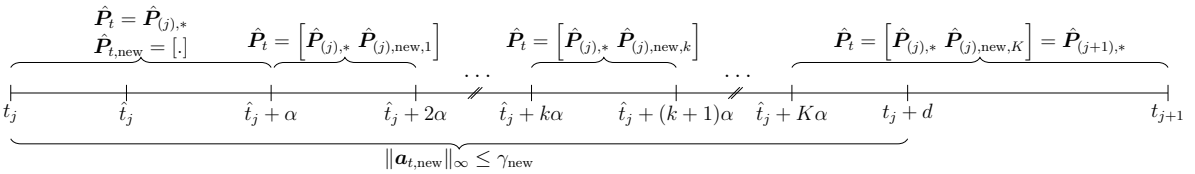

Figure 2.6: A diagram to visualize Algorithm 1 and Definition 2.6.4. The k-th projection-PCA step (at $t = \hat{t}_j + k\alpha$) computes the top left singular vectors of $(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')[\hat{\boldsymbol{\ell}}_{\hat{t}_j+(k-1)\alpha+1}, \hat{\boldsymbol{\ell}}_{\hat{t}_j+(k-1)\alpha+2}, \ldots \hat{\boldsymbol{\ell}}_{\hat{t}_j+k\alpha}]$.

3. At all times $\hat{\boldsymbol{P}}_t = [\hat{\boldsymbol{P}}_{t,*}\ \hat{\boldsymbol{P}}_{t,\text{new}}]$. Thus $\hat{\boldsymbol{P}}_t$ and $\hat{\boldsymbol{P}}_{t,\text{new}}$ update at every $t = \hat{t}_j + k\alpha$, $k = 1, 2, \ldots, K$, $j = 1, 2, \ldots, J$ while $\hat{\boldsymbol{P}}_{t,*}$ updates at every $t = \hat{t}_{j-1} + K\alpha$, $j = 2, \ldots, J$.

4. $\hat{\boldsymbol{P}}_{t-1,*} \perp \hat{\boldsymbol{P}}_{t,\text{new}}$ at $t = \hat{t}_j + k\alpha$ and so $\hat{\boldsymbol{P}}_{(j),*} \perp \hat{\boldsymbol{P}}_{(j),\text{new},k}$

5. $\boldsymbol{\Phi}_t = (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}' - \hat{\boldsymbol{P}}_{(j),\text{new},k}\hat{\boldsymbol{P}}_{(j),\text{new},k}{}')$ when $t \in \mathcal{J}_{\hat{u}_j+(k+1)}$, for $k = 1, 2, \ldots K-1$.

6. $\boldsymbol{\Phi}_t = (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')$ when $t \in [t_j, \hat{t}_j + \alpha]$ (recall that $\hat{t}_j = \hat{u}_j\alpha$).

7. $\boldsymbol{\Phi}_t = (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j+1),*}\hat{\boldsymbol{P}}_{(j+1),*}{}')$ when $t \in [\hat{t}_j + K\alpha + 1, t_{j+1} - 1]$.

Using the notation from the above definition, Figure 2.6 summarizes Algorithm 1.

Definition 2.6.5. *Recall that for basis matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$, $\text{dif}(\boldsymbol{P}, \boldsymbol{Q}) := \|(\boldsymbol{I} - \boldsymbol{P}\boldsymbol{P}')\boldsymbol{Q}\|_2$. Define*

1. $\zeta_{j,*} := \text{dif}(\hat{\boldsymbol{P}}_{(j),*}, \boldsymbol{P}_{(j),*})$

2. $\zeta_{j,\text{new},k} := \text{dif}([\hat{\boldsymbol{P}}_{(j),*}\ \hat{\boldsymbol{P}}_{(j),\text{new},k}], \boldsymbol{P}_{(j),\text{new}})$

Recall $\text{SE}_t = \text{dif}(\hat{\boldsymbol{P}}_t, \boldsymbol{P}_t)$. Notice that if subspace change times are correctly detected, for $t \in \mathcal{J}_{\hat{u}_j+k}$, $\text{SE}_t \le \zeta_{j,*} + \zeta_{j,\text{new},k-1}$ for $k = 1, 2, \ldots K$; for $t \in [t_j, \hat{t}_j + \alpha]$, $\text{SE}_t \le 1$; and for $t \in [\hat{t}_j + K\alpha + 1, t_{j+1} - 1]$, $\text{SE}_t = \zeta_{j+1,*}$.

Definition 2.6.6. *Define*

1. $\zeta^+_{j,*} := \big(r_0 + (j-1)r_{\text{new}}\big)\zeta$

2. $\zeta^+_{j,\text{new},0} := 1$, $\zeta^+_{j,\text{new},k} := \dfrac{b_{\boldsymbol{\mathcal{H}},k}}{b_{\boldsymbol{A}} - b_{\boldsymbol{A},\perp} - b_{\boldsymbol{\mathcal{H}},k}}$ *for* $k = 1, 2, \ldots, K$ *where* $b_{\boldsymbol{A}}$, $b_{\boldsymbol{A},\perp}$, *and* $b_{\boldsymbol{\mathcal{H}},k}$ *are defined in the remainder of this section. Their expressions are given by* (2.21), (2.22), *and* (2.23).

We will show that these are high probability upper bounds on $\zeta_{j,*}$ *and* $\zeta_{j,\text{new},k}$ *under appropriate conditioning.*

As we will see later, $b_{\boldsymbol{A}} \approx \lambda^-_{\text{new}}$, $b_{\boldsymbol{A},\perp} \approx {\zeta^+_{j,*}}^2 \lambda^+$ and $b_{\boldsymbol{\mathcal{H}},k} \approx 2\sqrt{\rho^2 h^+}\phi^+({\zeta^+_{j,*}}^2\lambda^+ + \zeta^+_{j,\text{new},k-1}\lambda^+_{\text{new}})$. Here $\approx$ means we are giving only the most dominant term for each expression. Thus,

$$\zeta^+_{j,\text{new},k} \approx \frac{2\sqrt{\rho^2 h^+}\phi^+(\zeta^+_{j,\text{new},k-1}\lambda^+_{\text{new}} + {\zeta^+_{j,*}}^2\lambda^+)}{\lambda^-_{\text{new}} - {\zeta^+_{j,*}}^2\lambda^+ - 2\sqrt{\rho^2 h^+}\phi^+(\zeta^+_{j,\text{new},k-1}\lambda^+_{\text{new}} + {\zeta^+_{j,*}}^2\lambda^+)}.$$

By using (2.5), the bounds on ζ from the theorem, and the bound on $\rho^2 h^+$, one can show that this decays roughly exponentially with k (see Lemma 2.6.14).

Definition 2.6.7. *Define the random variable*

$$X_u := \{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{u\alpha}\}$$

Definition 2.6.8. *Recall the definition of* $\boldsymbol{\mathcal{D}}_u$ *from Algorithm 1. For* $j = 1, \ldots, J$, $k = 1, \ldots, K$, *and for* $a = u_j$ *or* $a = u_j + 1$, *define the following events*

- $\text{DET}^a_j := \{\hat{u}_j = a\}$
- $\text{PPCA}^a_{j,k} := \left\{\hat{u}_j = a \text{ and } \text{rank}(\hat{\boldsymbol{P}}_{(j),\text{new},k}) = r_{j,\text{new}} \text{ and } \zeta_{j,\text{new},k} \le \zeta^+_{j,\text{new},k}\right\}$
- $\text{NODETS}^a_j := \left\{\hat{u}_j = a \text{ and } \lambda_{\max}\left(\frac{1}{\alpha}\boldsymbol{\mathcal{D}}_u \boldsymbol{\mathcal{D}}_u{}'\right) < \text{thresh for all } u \in [\hat{u}_j + K + 1, u_{j+1} - 1]\right\}$
- $\Gamma_{0,\text{end}} := \{\zeta_{1,*} \le r_0\zeta\} \cap \left\{\lambda_{\max}\left(\frac{1}{\alpha}\boldsymbol{\mathcal{D}}_u \boldsymbol{\mathcal{D}}_u{}'\right) < \text{thresh for all } u \in [1, u_1 - 1]\right\}$
- $\Gamma^a_{j,0} := \Gamma_{j-1,\text{end}} \cap \text{DET}^a_j$
- $\Gamma^a_{j,k} := \Gamma^a_{j,k-1} \cap \text{PPCA}^a_{j,k}$
- $\Gamma_{j,\text{end}} := \left(\Gamma^{u_j}_{j,K} \cap \text{NODETS}^{u_j}_j\right) \cup \left(\Gamma^{u_j+1}_{j,K} \cap \text{NODETS}^{u_j+1}_j\right)$

We misuse notation as follows. Suppose that a set Γ *is a subset of all possible values that a r.v.* X *can take. For two r.v.s'* $\{X, Y\}$, *when we need to say "*$X \in \Gamma$ *and* Y *can be anything"*

we will sometimes misuse notation and just say "$\{X, Y\} \in \Gamma$." For example, we sometimes say $X_{u_j} \in \Gamma_{j,\text{end}}$. This means $X_{u_j-1} \in \Gamma_{j,\text{end}}$ and $\boldsymbol{a}_t$ for $t \in \mathcal{J}_{u_j}$ are unconstrained.

Definition 2.6.9. *Define*

1. *Let $\boldsymbol{D}_{j,\text{new}} := (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{P}_{(j),\text{new}} \overset{QR}{=} \boldsymbol{E}_{j,\text{new}}\boldsymbol{R}_{j,\text{new}}$ denote its reduced QR decomposition, i.e. let $\boldsymbol{E}_{j,\text{new}}$ be a basis matrix for $\text{range}\,(\boldsymbol{D}_{j,\text{new}})$ and let $\boldsymbol{R}_{j,\text{new}} = \boldsymbol{E}_{j,\text{new}}{}'\boldsymbol{D}_{j,\text{new}}$.*

2. *Let $\boldsymbol{E}_{j,\text{new},\perp}$ be a basis matrix for the orthogonal complement of $\text{range}(\boldsymbol{E}_{j,\text{new}})$. To be precise, $\boldsymbol{E}_{j,\text{new},\perp}$ is an $n \times (n - r_j)$ basis matrix so that $[\boldsymbol{E}_{j,\text{new}}\ \boldsymbol{E}_{j,\text{new},\perp}]$ is unitary.*

3. *For $u = u_j + 1$ and $u = \hat{u}_j + k$ for $k = 1, \ldots, K$, define $\boldsymbol{A}_u$, $\boldsymbol{A}_{u,\perp}$, $\boldsymbol{\mathcal{A}}_u$ as*

$$\boldsymbol{A}_u := \frac{1}{\alpha}\sum_{t\in\mathcal{J}_u} \boldsymbol{E}_{j,\text{new}}{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{\ell}_t\boldsymbol{\ell}_t{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{E}_{j,\text{new}}$$

$$\boldsymbol{A}_{u,\perp} := \frac{1}{\alpha}\sum_{t\in\mathcal{J}_u} \boldsymbol{E}_{j,\text{new},\perp}{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{\ell}_t\boldsymbol{\ell}_t{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{E}_{j,\text{new},\perp}$$

and let

$$\boldsymbol{\mathcal{A}}_u := \begin{bmatrix} \boldsymbol{E}_{j,\text{new}} & \boldsymbol{E}_{j,\text{new},\perp} \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_u & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A}_{u,\perp} \end{bmatrix} \begin{bmatrix} \boldsymbol{E}_{j,\text{new}}{}' \\ \boldsymbol{E}_{j,\text{new},\perp}{}' \end{bmatrix}$$

4. *For $u = u_j + 1$ and $u = \hat{u}_j + k$ for $k = 1, \ldots, K$, define $\boldsymbol{\mathcal{M}}_u$ and $\boldsymbol{\mathcal{H}}_u$ as*

$$\boldsymbol{\mathcal{M}}_u = (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\left(\frac{1}{\alpha}\sum_{t\in\mathcal{J}_u} \hat{\boldsymbol{\ell}}_t\hat{\boldsymbol{\ell}}_t{}'\right)(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')$$

and

$$\boldsymbol{\mathcal{H}}_u := \boldsymbol{\mathcal{M}}_u - \boldsymbol{\mathcal{A}}_u$$

Remark 2.6.10. *Recall the definition of $\boldsymbol{\mathcal{D}}_u$ from Algorithm 1.*

Conditioned on $\Gamma_{j-1,\text{end}}$, for $u = u_j + 1$, $\hat{\boldsymbol{P}}_{u\alpha-1,*} = \hat{\boldsymbol{P}}_{(j),*}$ (in other words all $j-1$ previous subspace changes were detected) and thus, for this value of u,

$$\frac{1}{\alpha}\boldsymbol{\mathcal{D}}_u\boldsymbol{\mathcal{D}}_u{}' = \boldsymbol{\mathcal{M}}_u.$$

In this case, $\boldsymbol{\mathcal{M}}_u$ is the matrix whose maximum eigenvalue is checked to detect subspace change.

Conditioned on $\Gamma_{j,0}^{\hat{u}_j}$, for $u = \hat{u}_j + k$, $k = 1, 2, \ldots, K$, $\hat{\boldsymbol{P}}_{u\alpha-1,*} = \hat{\boldsymbol{P}}_{(j),*}$ and thus, for these values of u also,

$$\frac{1}{\alpha}\boldsymbol{\mathcal{D}}_u\boldsymbol{\mathcal{D}}_u{}' = \boldsymbol{\mathcal{M}}_u.$$

In this case, $\boldsymbol{\mathcal{M}}_u$ *is the matrix whose eigenvectors with eigenvalue above* thresh *form* $\hat{\boldsymbol{P}}_{(j),\text{new},k}$ *(see step 2 of Algorithm 1). In other words,* $\boldsymbol{\mathcal{M}}_u$ *has eigendecomposition*

$$\boldsymbol{\mathcal{M}}_u \overset{\text{EVD}}{=} \begin{bmatrix} \hat{\boldsymbol{P}}_{(j),\text{new},k} & \hat{\boldsymbol{P}}_{(j),\text{new},k,\perp} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\Lambda}}_u & \boldsymbol{0} \\ \boldsymbol{0} & \hat{\boldsymbol{\Lambda}}_{u,\perp} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{P}}_{(j),\text{new},k}{}' \\ \hat{\boldsymbol{P}}_{(j),\text{new},k,\perp}{}' \end{bmatrix}.$$

Definition 2.6.11. *Define*

1. $\kappa_{s,*} := \kappa_s(\boldsymbol{P}_{(J)})$ *and* $\kappa_{s,\text{new}} := \max_j \kappa_s(\boldsymbol{P}_{(j),\text{new}})$.
2. $\kappa_s^+ := 0.0215$. *As we will show later in Lemma 2.7.8, this upper bounds* $\|\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{D}_{j,\text{new}}\|_2$ *under appropriate conditioning.*
3. $\phi^+ := 1.2$. *As we will show later in Lemma 2.6.15, this upper bounds* $\phi_t := \|[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\|_2$ *under appropriate conditioning.*

Remark 2.6.12. *The entire proof uses Model 2.5.1 on* $\mathcal{T}_t$. *By Lemma 2.5.2, Model 2.2.3 is a special case of it. In particular, this means that (a) Model 2.2.3 also implies* $\rho^2 h^+ \leq 0.01$ *and (b) Model 2.2.3 also allows us to use Lemma 2.5.3. This lemma is used in the proof of Lemma 2.6.23 in Section 2.7.*

2.6.2 Five Main Lemmas for Proving Theorem 2.2.7

Fact 2.6.13. *Observe that* $\Gamma_{j,0}^a$ *both for* $a = u_j$ *and* $a = u_j + 1$ *implies that* $u_j \leq \hat{u}_j \leq u_j + 1$. *Thus, in both cases,* $t_j \leq \hat{t}_j \leq t_j + 2\alpha$. *So with the model assumption that* $d \geq (K+2)\alpha$, *we have that* $\mathcal{J}_{\hat{u}_j+k} \subseteq [t_j, t_j + d]$ *for* $k = 1, 2, \ldots, K$. *This fact is needed so that we can use the "slow subspace change" inequality, (2.5), to bound the eigenvalues along the new directions, and so that we can bound* $\|\boldsymbol{a}_{t,\text{new}}\|_\infty$ *by* γ_{new}.

Lemma 2.6.14. *[Exponential decay of the bound on* $\zeta_{j,\text{new},k}$ *(similar to [11, Lemma 6.1])] Under the conditions of Theorem 2.2.7,*

$$\zeta_{j,\text{new},k}^+ \leq 0.83^k + 0.84 r_{\text{new}} \zeta$$

This lemma follows by applying simple algebra on the definition and using the bounds assumed on ζ, λ_{new}^+ and $\rho^2 h^+$ in Theorem 2.2.7. A detailed proof of this lemma is given in Appendix 2.B.

Lemma 2.6.15 (Sparse Recovery Lemma (similar to [11, Lemma 6.4]))**.** *Assume that all of the conditions of Theorem 2.2.7 hold. Recall that* $\mathrm{SE}_t = \mathrm{dif}(\hat{\boldsymbol{P}}_t, \boldsymbol{P}_t)$.

1. *Conditioned on* $\Gamma_{j-1,\mathrm{end}}$*, for* $t \in [t_j, (\hat{u}_j + 1)\alpha]$

   (a) $\phi_t := \|[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\|_2 \leq \phi^+ := 1.2$.

   (b) *the support of* $\boldsymbol{x}_t$ *is recovered exactly i.e.* $\hat{\mathcal{T}}_t = \mathcal{T}_t$ *and* $\boldsymbol{e}_t$ *satisfies:*

   $$\boldsymbol{e}_t := \hat{\boldsymbol{x}}_t - \boldsymbol{x}_t = \boldsymbol{\ell}_t - \hat{\boldsymbol{\ell}}_t = \boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{\Phi}_t\boldsymbol{\ell}_t. \tag{2.18}$$

   (c) *Furthermore,*

   $$\mathrm{SE}_t \leq 1 \text{ , and}$$
   $$\|\boldsymbol{e}_t\|_2 \leq \phi^+(\zeta^+_{j,*}\sqrt{r}\gamma + \sqrt{r_{\mathrm{new}}}\gamma_{\mathrm{new}}) \leq 1.2\left(\sqrt{\zeta} + \sqrt{r_{\mathrm{new}}}\gamma_{\mathrm{new}}\right)$$

2. *For* $k = 2, 3, \ldots, K$ *and* $\hat{u}_j = u_j$ *or* $\hat{u}_j = u_j + 1$*, conditioned on* $\Gamma^{\hat{u}_j}_{j,k-1}$*, for* $t \in \mathcal{J}_{\hat{u}_j+k} = [(\hat{u}_j + k - 1)\alpha + 1, (\hat{u}_j + k)\alpha]$*, the first two conclusions above hold. That is,* $\phi_t \leq \phi^+$ *and* $\boldsymbol{e}_t$ *satisfies (2.18). Furthermore,*

   $$\mathrm{SE}_t \leq \zeta^+_{j,*} + \zeta^+_{j,\mathrm{new},k-1} \text{ , and}$$
   $$\|\boldsymbol{e}_t\|_2 \leq \phi^+(\zeta^+_{j,*}\sqrt{r}\gamma + \zeta^+_{j,\mathrm{new},k-1}\sqrt{r_{\mathrm{new}}}\gamma_{\mathrm{new}}) \leq 1.2\left(1.84\sqrt{\zeta} + (0.83)^{k-1}\sqrt{r_{\mathrm{new}}}\gamma_{\mathrm{new}}\right)$$

3. *For* $\hat{u}_j = u_j$ *or* $\hat{u}_j = u_j + 1$*, conditioned on* $\Gamma^{\hat{u}_j}_{j,K}$*, for* $t \in [(\hat{u}_j + K)\alpha + 1, t_{j+1} - 1]$*, the first two conclusions above hold (*$\phi_t \leq \phi^+$ *and* $\boldsymbol{e}_t$ *satisfies* (2.18)*). Furthermore,*

   $$\mathrm{SE}_t \leq \zeta^+_{j+1,*} \text{ , and}$$
   $$\|\boldsymbol{e}_t\|_2 \leq \phi^+\zeta^+_{j+1,*}\sqrt{r}\gamma \leq 1.2\sqrt{\zeta}$$

Notice that cases 1) and 3) of the above lemma occur when the algorithm is in the detection phase, while during the intervals for case 2) the algorithm is performing projection-PCA. In case 1) new directions have been added but not estimated, so the error is larger. In case 2), the error is decaying exponentially with each estimation step. Finally, case 3) occurs after the new directions have been successfully estimated and contains the tightest error bounds.

The proof is given in Appendix 2.C.

Lemma 2.6.16 (No false detection of subspace changes)**.**

1. *The event* $\Gamma_{j,K}^{\hat{u}_j}$ *and so also the event* $\Gamma_{j,\text{end}}$ *imply that* $\zeta_{j+1,*} \leq \zeta_{j+1,*}^{+}$.

2. $\mathbb{P}\left(\text{NODETS}_j^a \mid \Gamma_{j,K}^a\right) = 1$ *for* $a = u_j$ *or* $a = u_j + 1$.

Lemma 2.6.17 (Subspace change detected within 2α frames)**.** *For* $j = 1, \ldots, J$,

$$\mathbb{P}\left(\text{DET}^{u_j+1} \mid \Gamma_{j-1,\text{end}}, \overline{\text{DET}^{u_j}}\right) \geq p_{\text{det},1} := 1 - p_{\boldsymbol{A}} - p_{\boldsymbol{\mathcal{H}}}.$$

The definitions of $p_{\boldsymbol{A}}$ *and* $p_{\boldsymbol{\mathcal{H}}}$ *can be found in the remainder of this section.*

Lemma 2.6.18 (k-th iteration of pPCA works well)**.**

$$\mathbb{P}\left(\Gamma_{j,k}^a \mid \Gamma_{j,k-1}^a\right) = \mathbb{P}\left(\text{PPCA}_{j,k}^a \mid \Gamma_{j,k-1}^a\right) \geq p_{\text{ppca}} := 1 - p_{\boldsymbol{A}} - p_{\boldsymbol{A},\perp} - p_{\boldsymbol{\mathcal{H}}}$$

for $a = u_j$ *or* $a = u_j + 1$. *The definitions of* $p_{\boldsymbol{A}}$, $p_{\boldsymbol{A},\perp}$, *and* $p_{\boldsymbol{\mathcal{H}}}$ *can be found in the remainder of this section.*

The above lemma says that, conditioned on $k-1$ previous successful p-PCA steps and on accurate recovery of $P_{(j-1),*}$, the probability of correctly estimating $r_{j,\text{new}}$ and of a successful k^{th} projection PCA step is lower bounded by p_{ppca}. This is true whether the new directions are detected at u_j or at $u_j + 1$.

2.6.3 Proof of Theorem 2.2.7

Corollary 2.6.19. *The above lemmas imply that* $\mathbb{P}\left(\Gamma_{j,\text{end}} \mid \Gamma_{j-1,\text{end}}\right) \geq p_{\text{det},1} \cdot (p_{\text{ppca}})^K$.

Proof. Let

$$p_{\text{det},0} := \mathbb{P}\left(\text{DET}^{u_j} \mid \Gamma_{j-1,\text{end}}\right).$$

From the above lemmas, we get that

$$\begin{aligned}
\mathbb{P}\left(\Gamma_{j,\text{end}} \mid \Gamma_{j-1,\text{end}}\right) &= \mathbb{P}\Big(\left(\text{DET}^{u_j} \cap \text{PPCA}_{j,1}^{u_j} \cap \cdots \cap \text{PPCA}_{j,K}^{u_j} \cap \text{NODETS}_{j}^{u_j}\right) \cup \\
&\qquad \left(\overline{\text{DET}^{u_j}} \cap \text{DET}^{u_j+1} \cap \text{PPCA}_{j,k}^{u_j+1} \cap \cdots \cap \text{PPCA}_{j,K}^{u_j+1} \cap \text{NODETS}_{j}^{u_j+1}\right) \mid \Gamma_{j-1,\text{end}}\Big) \\
&= \mathbb{P}\left(\text{DET}^{u_j} \cap \text{PPCA}_{j,1}^{u_j} \cap \cdots \cap \text{PPCA}_{j,K}^{u_j} \mid \Gamma_{j-1,\text{end}}\right) \\
&\quad + \mathbb{P}\left(\overline{\text{DET}^{u_j}} \cap \text{DET}^{u_j+1} \cap \text{PPCA}_{j,k}^{u_j+1} \cap \cdots \cap \text{PPCA}_{j,K}^{u_j+1} \mid \Gamma_{j-1,\text{end}}\right) \\
&\geq p_{\text{det},0} \cdot (p_{\text{ppca}})^K + (1 - p_{\text{det},0}) \cdot p_{\text{det},1} \cdot (p_{\text{ppca}})^K \\
&\geq p_{\text{det},0} \cdot p_{\text{det},1} \cdot (p_{\text{ppca}})^K + (1 - p_{\text{det},0}) \cdot p_{\text{det},1} \cdot (p_{\text{ppca}})^K \\
&= p_{\text{det},1} \cdot (p_{\text{ppca}})^K.
\end{aligned}$$

⊠

Proof of Theorem 2.2.7. Theorem 2.2.7 follows from Corollary 2.6.19 and the assumed lower bound on α. Notice that by Lemma 2.6.14, the choice of K, and Lemma 2.6.15, the event $\Gamma_{J,\text{end}}$ will imply all conclusions of the theorem.

By the first assumption (accurate initial subspace knowledge) and the argument used to prove Lemma 2.6.16, we get that $\mathbb{P}(\Gamma_{0,\text{end}}) = 1$. By the chain rule,

$$\mathbb{P}(\Gamma_{J,\text{end}}) = \prod_{j=1}^{J} \mathbb{P}(\Gamma_{j,\text{end}} \mid \Gamma_{j-1,\text{end}}, \Gamma_{j-2,\text{end}}, \ldots, \Gamma_{1,\text{end}}, \Gamma_{0,\text{end}}).$$

Because $\Gamma_{j-1,\text{end}} \subseteq \Gamma_{j-2,\text{end}} \subseteq \cdots \subseteq \Gamma_{1,\text{end}} \subseteq \Gamma_{0,\text{end}}$, we get

$$\begin{aligned}
\mathbb{P}(\Gamma_{J,\text{end}}) &= \prod_{j=1}^{J} \mathbb{P}(\Gamma_{j,\text{end}} \mid \Gamma_{j-1,\text{end}}) \\
&\geq \prod_{j=1}^{J} p_{\text{det},1} \cdot (p_{\text{ppca}})^K = (p_{\text{det},1} \cdot (p_{\text{ppca}})^K)^J \\
&\geq 1 - n^{-10}
\end{aligned}$$

The last line is by the lower bound on α assumed in the theorem and the fact that $p_{\text{det},1} \geq p_{\text{ppca}}$. ⊠

2.6.4 Key Lemmas for Proving of Lemmas 2.6.16, 2.6.17, and 2.6.18

Before proving the lemmas from the preceding subsection, we introduce several lemmas which will be used in the proofs.

The following lemma follows from the $\sin\theta$ theorem [22] and Weyl's theorem. It is taken from [11].

Lemma 2.6.20 ([11], Lemma 6.9)**.** *At $u = \hat{u}_j + k$, if $\operatorname{rank}(\hat{\boldsymbol{P}}_{(j),\text{new},k}) = r_{j,\text{new}}$, and if $\lambda_{\min}(\boldsymbol{A}_u) - \|\boldsymbol{A}_{u,\perp}\|_2 - \|\boldsymbol{\mathcal{H}}_u\|_2 > 0$, then*

$$\zeta_{j,\text{new},k} \leq \frac{\|\boldsymbol{\mathcal{H}}_u\|_2}{\lambda_{\min}(\boldsymbol{A}_u) - \|\boldsymbol{A}_{u,\perp}\|_2 - \|\boldsymbol{\mathcal{H}}_u\|_2} \tag{2.19}$$

The next three lemmas each assert a high probability bound for one of the terms in (2.19). In the following lemmas, let

$$\epsilon = \frac{r_{\text{new}}\zeta\hat{\lambda}^-_{\text{train}}}{100}. \tag{2.20}$$

Let $p_{\boldsymbol{A}} := r_{\text{new}} \exp\left(\frac{-\alpha\zeta^2(\hat{\lambda}^-_{\text{train}})^2}{8\cdot 100^2\cdot {\gamma_{\text{new}}}^4}\right) + r_{\text{new}} \exp\left(\frac{-\alpha {r_{\text{new}}}^2\zeta^2(\hat{\lambda}^-_{\text{train}})^2}{8\cdot 100^2\cdot 4^2}\right)$ and

$$b_{\boldsymbol{A}} := (1 - (\zeta^+_{j,*})^2)\lambda^-_{\text{new}} - 2\epsilon. \tag{2.21}$$

Lemma 2.6.21. *For $k = 1, \ldots, K$,*

$$\mathbb{P}\left(\lambda_{\min}\left(\boldsymbol{A}_{\hat{u}_j+k}\right) \geq b_{\boldsymbol{A}} \;\middle|\; X_{\hat{u}_j+k-1}\right) \geq 1 - p_{\boldsymbol{A}}$$

for all $X_{\hat{u}_j+k-1} \in \Gamma^{\hat{u}_j}_{j,k-1}$ with $\hat{u}_j = u_j$ or $\hat{u}_j = u_j + 1$.

The same bound holds for $\lambda_{\min}(\boldsymbol{A}_{u_j+1})$ when we condition on $X_{u_j} \in \Gamma_{j-1,\text{end}}$.

Let $p_{\boldsymbol{A},\perp} := (n - r_{\text{new}}) \exp\left(\frac{-\alpha {r_{\text{new}}}^2\zeta^2(\hat{\lambda}^-_{\text{train}})^2}{8\cdot 100^2}\right)$

and

$$b_{\boldsymbol{A},\perp} := (\zeta^+_{j,*})^2\lambda^+ + \epsilon. \tag{2.22}$$

Lemma 2.6.22. *For $k = 1, \ldots, K$,*

$$\mathbb{P}\left(\lambda_{\max}\left(\boldsymbol{A}_{\hat{u}_j+k,\perp}\right) \leq b_{\boldsymbol{A},\perp} \;\middle|\; X_{\hat{u}_j+k-1}\right) \geq 1 - p_{\boldsymbol{A},\perp}$$

for all $X_{\hat{u}_j+k-1} \in \Gamma^{\hat{u}_j}_{j,k-1}$ with $\hat{u}_j = u_j$ or $\hat{u}_j = u_j + 1$.

The same bound holds for $\lambda_{\max}(\boldsymbol{A}_{u_j+1,\perp})$ when we condition on $X_{u_j} \in \Gamma_{j-1,\text{end}}$.

Let

$$p_{\boldsymbol{\mathcal{H}}} := n \exp\left(\frac{-\alpha {r_{\text{new}}}^2 \zeta^2 (\hat{\lambda}^-_{\text{train}})^2}{32 \cdot 100^2 (\phi^+)^2 (\sqrt{\zeta} + \sqrt{r_{\text{new}}}\gamma_{\text{new}})^4}\right) + n \exp\left(\frac{-\alpha {r_{\text{new}}}^2 \zeta^2 (\hat{\lambda}^-_{\text{train}})^2}{8 \cdot 100^2 \left(\phi^+ (\sqrt{\zeta} + \sqrt{r_{\text{new}}}\gamma_{\text{new}})\right)^4}\right) + n \exp\left(\frac{-\alpha {r_{\text{new}}}^2 \zeta^2 (\hat{\lambda}^-_{\text{train}})^2}{32 \cdot 100^2 (\zeta + \sqrt{\zeta}\sqrt{r_{\text{new}}}\gamma_{\text{new}})^2}\right).$$

and

$$b_{\boldsymbol{\mathcal{H}},k} := 2b_{\boldsymbol{\ell e},k} + b_{\boldsymbol{ee},k} + 2b_{\boldsymbol{F}} \tag{2.23}$$

where

$$b_{\boldsymbol{\ell e},k} := \begin{cases} \phi^+\left(\sqrt{\rho^2 h^+}(\zeta^+_{j,*})^2\lambda^+ + \kappa_s^+\lambda^+_{\text{new}}\right) + \epsilon & k = 1 \\ \left[(\zeta^+_{j,*})^2\lambda^+ + \zeta^+_{j,\text{new},k-1}\lambda^+_{\text{new}}\right]\left(\sqrt{\rho^2 h^+}\phi^+\right) + \epsilon & k \geq 2 \end{cases}$$

$$b_{\boldsymbol{ee},k} := \begin{cases} \rho^2 h^+ (\phi^+)^2\left((\zeta^+_{j,*})^2\lambda^+ + (\kappa_s^+)^2\lambda^+_{\text{new}}\right) + \epsilon & k = 1 \\ \rho^2 h^+ (\phi^+)^2\left((\zeta^+_{j,*})^2(\lambda^+) + (\zeta^+_{j,\text{new},k-1})^2(\lambda^+_{\text{new}})\right) + \epsilon & k \geq 2 \end{cases}$$

and

$$b_{\boldsymbol{F}} := (\zeta^+_{j,*})^2\lambda^+ + \epsilon.$$

Lemma 2.6.23. *For $k = 1, \ldots, K$,*

$$\mathbb{P}\left(\|\boldsymbol{\mathcal{H}}_{\hat{u}_j+k}\|_2 \leq b_{\boldsymbol{\mathcal{H}},k} \;\middle|\; X_{\hat{u}_j+k-1}\right) \geq 1 - p_{\boldsymbol{\mathcal{H}}} \tag{2.24}$$

for all $X_{\hat{u}_j+k-1} \in \Gamma^{\hat{u}_j}_{j,k-1}$ with $\hat{u}_j = u_j$ or $\hat{u}_j = u_j + 1$

The same bound ($k = 1$ case), i.e. $\|\boldsymbol{\mathcal{H}}_{u_j+1}\|_2 \leq b_{\boldsymbol{\mathcal{H}},1}$, also holds with the same probability when we condition on $X_{u_j} \in \Gamma_{j-1,\text{end}}$.

The above lemmas are proved in the next section (Section 2.7). The proofs use Fact 2.6.13.

2.6.5 Proofs of Lemmas 2.6.16, 2.6.17, and 2.6.18

Proof of Lemma 2.6.16. Recall that $\Gamma_{j,\text{end}} := \left(\Gamma^{u_j}_{j,K} \cap \text{NODETS}^{u_j}_j\right) \cup \left(\Gamma^{u_j+1}_{j,K} \cap \text{NODETS}^{u_j+1}_j\right)$.

1. By the definition of $\Gamma_{j,K}^{\hat{u}_j}$, both for $\hat{u}_j = u_j$ and $\hat{u}_j = u_j+1$, $\zeta_{j,*} \le \zeta_{j,*}^+ = (r_0+(j-1)r_{\text{new}})\zeta$ and $\zeta_{j,K} \le \zeta_{j,\text{new},K}^+$. Lemma 2.6.14 and the choice of K imply that $\zeta_{j,\text{new},K}^+ \le r_{\text{new}}\zeta$. Thus, $\zeta_{j+1,*} \le \zeta_{j,*} + \zeta_{j,\text{new},k} \le \zeta_{j+1,*}^+ = (r_0 + jr_{\text{new}})\zeta$.

2. $\mathbb{P}(\text{NODETS}_j^{\hat{u}_j} \mid \Gamma_{j,K}^{\hat{u}_j}) = \mathbb{P}\Big(\lambda_{\max}(\frac{1}{\alpha}\boldsymbol{\mathcal{D}}_u\boldsymbol{\mathcal{D}}_u) < \text{thresh for all } u \in [\hat{u}_j+K+1, u_{j+1}-1] \mid \Gamma_{j,K}^{\hat{u}_j}\Big)$ for $\hat{u}_j = u_j$ or $\hat{u}_j = u_j + 1$.

As shown in 1), $\Gamma_{j,K}^{\hat{u}_j}$ implies that $\text{dif}(\hat{\boldsymbol{P}}_{(j+1),*}, \boldsymbol{P}_{(j+1),*}) \le \zeta_{j+1,*}^+ = (r_0 + jr_{\text{new}})\zeta$. Recall that $\boldsymbol{P}_{(j+1),*} = \boldsymbol{P}_{(j)}$. Also, for $u \in [\hat{u}_j + K + 1, u_{j+1} - 1]$, $\hat{\boldsymbol{P}}_{u\alpha-1,*} = \hat{\boldsymbol{P}}_{(j+1),*}$. Also, for all $t \in \mathcal{J}_u$ for these u's, $\boldsymbol{\ell}_t = \boldsymbol{P}_{(j)}\boldsymbol{a}_t = \boldsymbol{P}_{(j+1),*}\boldsymbol{a}_t$. Therefore,

$$\begin{aligned}
\lambda_{\max}\left(\frac{1}{\alpha}\boldsymbol{\mathcal{D}}_u\boldsymbol{\mathcal{D}}_u\right) &= \lambda_{\max}\left(\frac{1}{\alpha}\sum_{t\in\mathcal{J}_u}(\boldsymbol{I} - \hat{\boldsymbol{P}}_{u\alpha-1,*}\hat{\boldsymbol{P}}_{u\alpha-1,*}{}')\hat{\boldsymbol{\ell}}_t\hat{\boldsymbol{\ell}}_t{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{u\alpha-1,*}\hat{\boldsymbol{P}}_{u\alpha-1,*}{}')\right) \\
&= \lambda_{\max}\Bigg(\frac{1}{\alpha}\sum_{t\in\mathcal{J}_u}(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j+1),*}\hat{\boldsymbol{P}}_{(j+1),*}{}')(\boldsymbol{P}_{(j)}\boldsymbol{a}_t - \boldsymbol{e}_t) \\
&\qquad\qquad (\boldsymbol{P}_{(j)}\boldsymbol{a}_t - \boldsymbol{e}_t)'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j+1),*}\hat{\boldsymbol{P}}_{(j+1),*}{}')\Bigg) \\
&\le (\zeta_{j+1,*}^+)^2 r\gamma^2 + 2\phi^+(\zeta_{j+1,*}^+)^2 r\gamma^2 + (\phi^+)^2(\zeta_{j+1,*}^+)^2 r\gamma^2 \\
&\le 4(\phi^+)^2\zeta\hat{\lambda}_{\text{train}}^- \le \frac{\hat{\lambda}_{\text{train}}^-}{2}.
\end{aligned}$$

The bound on $\boldsymbol{e}_t$ comes from Lemma 2.6.15. The penultimate inequality uses the bound $\zeta \le \frac{\lambda_{\text{train}}^-}{r^3\gamma^2}$ assumed in Theorem 2.2.7. ⊠

The next two proofs follow using the following two facts and the four lemmas from the previous subsection.

Fact 2.6.24. *For an event $\mathcal{E}$ and random variable X, $\mathbb{P}(\mathcal{E}|X) \ge p$ for all $X \in \mathcal{C}$ implies that $\mathbb{P}(\mathcal{E}|X \in \mathcal{C}) \ge p$.*

Fact 2.6.25. *Using the bounds on ζ and on $\rho^2 h^+$ and using (3b), we get*

$$\begin{aligned}
b_{\boldsymbol{A}} &\ge 0.94\lambda_{\text{new}}^- \ge 0.94\hat{\lambda}_{\text{train}}^- \\
b_{\boldsymbol{A},\perp} &\le 0.011\hat{\lambda}_{\text{train}}^- \\
b_{\boldsymbol{\mathcal{H}},k} &\le 0.24\hat{\lambda}_{\text{train}}^-.
\end{aligned}$$

Thus, $b_{\boldsymbol{A}} - b_{\boldsymbol{\mathcal{H}},k} \ge 0.5\hat{\lambda}_{\text{train}}^- = \text{thresh}$ and $b_{\boldsymbol{A},\perp} + b_{\boldsymbol{\mathcal{H}},k} < 0.25\hat{\lambda}_{\text{train}}^- < \text{thresh}$.

Proof of Lemma 2.6.17. We will prove that $\mathbb{P}\left(\mathrm{DET}^{u_j+1} \mid X_{u_j}\right) > p_{\mathrm{det},1}$ for all $X_{u_j} \in \Gamma_{j-1,\mathrm{end}}$. In particular, this will imply that $\mathbb{P}(\mathrm{DET}^{u_j+1} \mid X_{u_j}) > p_{\mathrm{det},1}$ for all $X_{u_j} \in \Gamma_{j-1,\mathrm{end}} \cap \overline{\mathrm{DET}^{u_j}}$ and so we can conclude that $\mathbb{P}(\mathrm{DET}^{u_j+1} \mid \Gamma_{j-1,\mathrm{end}}, \overline{\mathrm{DET}^{u_j}}) > p_{\mathrm{det},1}$.

Recall that $\boldsymbol{\mathcal{M}}_u = \frac{1}{\alpha}\boldsymbol{\mathcal{D}}_u\boldsymbol{\mathcal{D}}_u{}'$, and observe that

$$\mathbb{P}\left(\mathrm{DET}^{u_j+1} \mid X_{u_j}\right) = \mathbb{P}\left(\lambda_{\max}(\boldsymbol{\mathcal{M}}_{u_j+1}) > \mathrm{thresh} \mid X_{u_j}\right)$$

By Weyl's Theorem

$$\begin{aligned}
\lambda_{\max}(\boldsymbol{\mathcal{M}}_{u_j+1}) &\geq \lambda_{\max}(\boldsymbol{\mathcal{A}}_{u_j+1}) + \lambda_{\min}(\boldsymbol{\mathcal{H}}_{u_j+1}) \\
&\geq \lambda_{\max}(\boldsymbol{A}_{u_j+1}) - \|\boldsymbol{\mathcal{H}}_{u_j+1}\|_2 \\
&\geq \lambda_{\min}(\boldsymbol{A}_{u_j+1}) - \|\boldsymbol{\mathcal{H}}_{u_j+1}\|_2
\end{aligned}$$

When $X_{u_j} \in \Gamma_{j-1,\mathrm{end}}$, Lemmas 2.6.21 and 2.6.23 applied with ϵ given by (2.20) show that $\lambda_{\min}(\boldsymbol{A}_{u_j+1}) \geq b_{\boldsymbol{A}}$ and $\|\boldsymbol{\mathcal{H}}_{u_j+1}\|_2 \leq b_{\boldsymbol{\mathcal{H}},1}$ with probability at least $1 - p_{\boldsymbol{A}} - p_{\boldsymbol{\mathcal{H}}} = p_{\mathrm{det},1}$. Using Fact 2.6.25, $b_{\boldsymbol{A}} - b_{\boldsymbol{\mathcal{H}},1} \geq \mathrm{thresh}$ and so the lemma follows. ⊠

Proof of Lemma 2.6.18. To prove this Lemma we need to show two things. First, conditioned on $\Gamma_{j,k-1}^{\hat{u}_j}$, the k^{th} estimate of the number of new directions is correct. That is: $\hat{r}_{j,\mathrm{new},k} = r_{j,\mathrm{new}}$. Second, we must show $\zeta_{j,\mathrm{new},k} \leq \zeta^+_{j,\mathrm{new},k}$, again conditioned on $\Gamma_{j,k-1}^{\hat{u}_j}$.

Notice that $\hat{r}_{j,\mathrm{new},k} = \mathrm{rank}(\hat{\boldsymbol{P}}_{(j),\mathrm{new},k})$. To show that $\mathrm{rank}(\hat{\boldsymbol{P}}_{(j),\mathrm{new},k}) = r_{j,\mathrm{new}}$, we need to show that for $u = \hat{u}_j + k$, $k = 1, \ldots, K$, $\lambda_{r_{j,\mathrm{new}}}(\boldsymbol{\mathcal{M}}_u) > \mathrm{thresh}$ and $\lambda_{r_{j,\mathrm{new}}+1}(\boldsymbol{\mathcal{M}}_u) < \mathrm{thresh}$. To do this we proceed similarly to above.

Observe that, $\boldsymbol{\mathcal{M}}_u = \boldsymbol{\mathcal{A}}_u + \boldsymbol{\mathcal{H}}_u$. By Fact 2.6.25, $b_{\boldsymbol{A}} > b_{\boldsymbol{A},\perp}$. Combining this with Lemmas 2.6.21 and 2.6.22 gives, $\lambda_{\min}(\boldsymbol{A}_u) > \lambda_{\max}(\boldsymbol{A}_{u,\perp})$ with probability at least $1 - p_{\boldsymbol{A}} - p_{\boldsymbol{A},\perp}$ under the appropriate conditioning (conditioned on $\Gamma_{j,k-1}^{\hat{u}_j}$). Since $\boldsymbol{A}_u$ is of size $r_{j,\mathrm{new}} \times r_{j,\mathrm{new}}$, this means that $\lambda_{r_{j,\mathrm{new}}}(\boldsymbol{\mathcal{A}}_u) = \lambda_{\min}(\boldsymbol{A}_u)$ and $\lambda_{r_{j,\mathrm{new}}+1}(\boldsymbol{\mathcal{A}}_u) = \lambda_{\max}(\boldsymbol{A}_{u,\perp})$. Using this and Weyl's Theorem,

$$\begin{aligned}
\lambda_{r_{j,\mathrm{new}}}(\boldsymbol{\mathcal{M}}_u) &\geq \lambda_{r_{j,\mathrm{new}}}(\boldsymbol{\mathcal{A}}_u) + \lambda_{\min}(\boldsymbol{\mathcal{H}}_u) \\
&\geq \lambda_{r_{j,\mathrm{new}}}(\boldsymbol{\mathcal{A}}_u) - \|\boldsymbol{\mathcal{H}}_u\|_2 \\
&= \lambda_{\min}(\boldsymbol{A}_u) - \|\boldsymbol{\mathcal{H}}_u\|_2
\end{aligned}$$

and

$$\begin{aligned}\lambda_{r_{j,\text{new}}+1}(\boldsymbol{\mathcal{M}}_u) &\leq \lambda_{r_{j,\text{new}}+1}(\boldsymbol{\mathcal{A}}_u) + \lambda_{\max}(\boldsymbol{\mathcal{H}}_u) \\ &\leq \lambda_{r_{j,\text{new}}+1}(\boldsymbol{\mathcal{A}}_u) + \|\boldsymbol{\mathcal{H}}_u\|_2 \\ &= \lambda_{\max}(\boldsymbol{\mathcal{A}}_{u,\perp}) + \|\boldsymbol{\mathcal{H}}_u\|_2\end{aligned}$$

with probability at least $1 - p_{\boldsymbol{A}} - p_{\boldsymbol{A},\perp}$ under the appropriate conditioning. Using Lemmas 2.6.21, 2.6.22, and 2.6.23 applied with ϵ given by (2.20) and Fact 2.6.25, we can conclude that with probability greater than p_{ppca}, $\lambda_{r_{j,\text{new}}}(\boldsymbol{\mathcal{M}}_u) > b_{\boldsymbol{A}} - b_{\boldsymbol{\mathcal{H}},k} \geq \text{thresh}$ and $\lambda_{r_{j,\text{new}}+1}(\boldsymbol{\mathcal{M}}_u) < b_{\boldsymbol{A},\perp} + b_{\boldsymbol{\mathcal{H}},k} < \text{thresh}$. Therefore $\text{rank}(\hat{\boldsymbol{P}}_{(j),\text{new},k}) = r_{j,\text{new}}$ with probability greater than p_{ppca} under the appropriate conditioning.

To show that $\zeta_{j,\text{new},k} \leq \zeta^+_{j,\text{new},k}$, we also use Lemmas 2.6.21, 2.6.22, and 2.6.23 applied with ϵ given by (2.20). Using $\text{rank}(\hat{\boldsymbol{P}}_{(j),\text{new},k}) = r_{j,\text{new}}$ and applying Lemma 2.6.20 with these bounds gives the desired result. ⊠

2.7 Proofs of Lemmas 2.6.21, 2.6.22, and 2.6.23

2.7.1 Some definitions, remarks and facts

Definition 2.7.1. *Define the following for $k = 0, 1, \ldots, K$. Recall that $\hat{\boldsymbol{P}}_{(j),\text{new},0} = [.]$.*

1. $\boldsymbol{D}_{j,\text{new},k} := (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}' - \hat{\boldsymbol{P}}_{(j),\text{new},k}\hat{\boldsymbol{P}}_{(j),\text{new},k}{}')\boldsymbol{P}_{(j),\text{new}}$. *Thus* $\boldsymbol{D}_{j,\text{new}} = \boldsymbol{D}_{j,\text{new},0}$.
2. $\boldsymbol{D}_{j,*,k} := (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}' - \hat{\boldsymbol{P}}_{(j),\text{new},k}\hat{\boldsymbol{P}}_{(j),\text{new},k}{}')\boldsymbol{P}_{(j),*}$ *and* $\boldsymbol{D}_{j,*} := \boldsymbol{D}_{j,*,0}$.
3. *Recall that* $\zeta_{j,\text{new},0} = \|\boldsymbol{D}_{j,\text{new}}\|_2$, $\zeta_{j,\text{new},k} = \|\boldsymbol{D}_{j,\text{new},k}\|_2$, $\zeta_{j,*} = \|\boldsymbol{D}_{j,*}\|_2$. *Also, clearly,* $\|\boldsymbol{D}_{j,*,k}\|_2 \leq \|\boldsymbol{D}_{j,*}\|_2 \leq \zeta_{j,*}$.

Definition 2.7.2. *For ease of notation, define*

$$\tilde{\boldsymbol{\ell}}_t := (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{\ell}_t$$

Remark 2.7.3. *In the rest of this section, for ease of notation, we do the following.*

- *We remove the subscript j from $\boldsymbol{D}_{j,\text{new},k}$, $\boldsymbol{E}_{j,\text{new}}$, and $\zeta_{j,\text{new},k}$ etc. and from everything in Definitions 2.6.3, 2.6.4, 2.6.5, 2.6.9 and 2.7.1.*

- *Similarly we also let* $X_k := X_{\hat{u}_j+k}$ *and* $\Gamma_k := \Gamma_{j,k}^{\hat{u}_j}$ *for both* $\hat{u}_j = u_j$ *and* $\hat{u}_j = u_{j+1}$. *More precisely, whenever we say* $\mathbb{P}\left(\text{event}|X_{k-1} \in \Gamma_{k-1}\right) \geq p_0$ *we mean* $\mathbb{P}\left(\text{event}|X_{u_j+k-1} \in \Gamma_{j,k-1}^{u_j}\right) \geq p_0$ *and* $\mathbb{P}\left(\text{event}|X_{u_j+1+k-1} \in \Gamma_{j,k-1}^{u_j+1}\right) \geq p_0$.
- *Finally,* $\sum_t$ *refers to* $\sum_{t\in\mathcal{J}_u}$ *for* $u = \hat{u}_j + k$

Also, note the following.

- *The proof for the bound on* $\boldsymbol{A}_u$ *for* $u = u_j + 1$ *is the same as that for* $u = \hat{u}_j + 1$ *since in both cases* $\hat{\boldsymbol{P}}_{t,*} = \hat{\boldsymbol{P}}_{(j),*}$ *and* $\hat{\boldsymbol{P}}_{t,\text{new}} = [.]$ *for all* $t \in \mathcal{J}_u$. *The same is true for the bounds on* $\boldsymbol{A}_{u_j+1,\perp}$ *and* $\boldsymbol{\mathcal{H}}_{u_j+1}$.

Fact 2.7.4. *When* $X_{k-1} \in \Gamma_{k-1}$,

1. $\|\boldsymbol{D}_{*,k-1}\|_2 \leq \zeta_{j,*}^+$ *for* $k = 1, \ldots, K$.
2. $\|\boldsymbol{D}_{\text{new},k-1}\|_2 \leq \zeta_{\text{new},k-1}^+$ *for* $k = 1, \ldots, K+1$ *(by definition of* Γ_{k-1}*).*
3. *Recall that* $\zeta_{\text{new},0}^+ = 1$.
4. $\|[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\|_2 \leq \phi^+$ *(from Lemma 2.6.15)*
5. $\lambda_{\min}(\boldsymbol{R}_{\text{new}}\boldsymbol{R}_{\text{new}}{}') \geq 1-(\zeta_*^+)^2$ *(this follows because* $\|\hat{\boldsymbol{P}}_*{}'\boldsymbol{P}_{\text{new}}\|_2 = \|\hat{\boldsymbol{P}}_*(\boldsymbol{I}-\boldsymbol{P}_*\boldsymbol{P}_*{}')'\boldsymbol{P}_{\text{new}}\|_2 \leq \zeta_*$*)*
6. $\boldsymbol{E}_{\text{new}}{}'\boldsymbol{D}_{\text{new}} = \boldsymbol{E}_{\text{new}}{}'\boldsymbol{E}_{\text{new}}\boldsymbol{R}_{\text{new}} = \boldsymbol{R}_{\text{new}}$ *and* $\boldsymbol{E}_{\text{new},\perp}{}'\boldsymbol{D}_{\text{new}} = \boldsymbol{0}$.
7. $\tilde{\boldsymbol{\ell}}_t = \boldsymbol{D}_*\boldsymbol{a}_{t,*} + \boldsymbol{D}_{\text{new}}\boldsymbol{a}_{t,\text{new}}$.
8. $\boldsymbol{e}_t$ *satisfies (2.18) with probability one, i.e.* $\boldsymbol{e}_t = \boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'(\boldsymbol{D}_{*,k-1}\boldsymbol{a}_{t,*} + \boldsymbol{D}_{\text{new},k-1}\boldsymbol{a}_{t,\text{new}})$.

2.7.2 Preliminaries

First observe that the matrices $\boldsymbol{D}_{\text{new}}$, $\boldsymbol{R}_{\text{new}}$, $\boldsymbol{E}_{\text{new}}$, $\boldsymbol{D}_*, \boldsymbol{D}_{\text{new},k-1}$ are all functions of the random variable X_{k-1}. Since X_{k-1} is independent of any $\boldsymbol{a}_t$ for $t \in \mathcal{J}_{\hat{u}_j+k}$, the same is true for these matrices. All terms that we bound for Lemmas 2.6.21 and 2.6.22 are of the form

$\frac{1}{\alpha}\sum_{t\in\mathcal{J}_{\hat{u}_j+k}} \boldsymbol{Z}_t$ where $\boldsymbol{Z}_t = f_1(X_{k-1})\boldsymbol{Y}_t f_2(X_{k-1})$, $\boldsymbol{Y}_t$ is a sub-matrix of $\boldsymbol{a}_t\boldsymbol{a}_t'$, and $f_1(.)$ and $f_2(.)$ are functions of X_{k-1}. Thus, conditioned on X_{k-1}, the $\boldsymbol{Z}_t$'s are mutually independent.

All the terms that we bound for Lemma 2.6.23 contain $\boldsymbol{e}_t$. Using Lemma 2.6.15, conditioned on X_{k-1}, $\boldsymbol{e}_t$ satisfies (2.18) with probability one whenever $X_{k-1} \in \Gamma_{k-1}$. Using (2.18), it is easy to see that all the terms needed for this lemma are also of the above form whenever $X_{k-1} \in \Gamma_{k-1}$. Thus, conditioned on X_{k-1}, the $\boldsymbol{Z}_t$'s for all the above terms are mutually independent, whenever $X_{k-1} \in \Gamma_{k-1}$.

We will use the following corollaries of the matrix Hoeffding inequality from [23]. These are proved in [11].

Corollary 2.7.5 (Matrix Hoeffding conditioned on another random variable for a nonzero mean Hermitian matrix [23, 11])**.** *Given an α-length sequence $\{\boldsymbol{Z}_t\}$ of random Hermitian matrices of size $n \times n$, a r.v. X, and a set $\mathcal{C}$ of values that X can take. Assume that, for all $X \in \mathcal{C}$, (i) $\boldsymbol{Z}_t$'s are conditionally independent given X; (ii) $\mathbb{P}(b_1\boldsymbol{I} \preceq \boldsymbol{Z}_t \preceq b_2\boldsymbol{I}|X) = 1$ and (iii) $b_3\boldsymbol{I} \preceq \frac{1}{\alpha}\sum_t \mathbb{E}(\boldsymbol{Z}_t|X) \preceq b_4\boldsymbol{I}$. Then for all $\epsilon > 0$,*

$$\mathbb{P}\left(\lambda_{\max}\left(\frac{1}{\alpha}\sum_t \boldsymbol{Z}_t\right) \leq b_4 + \epsilon \middle| X\right) \geq 1 - n\exp\left(\frac{-\alpha\epsilon^2}{8(b_2-b_1)^2}\right) \text{ for all } X \in \mathcal{C}$$

$$\mathbb{P}\left(\lambda_{\min}\left(\frac{1}{\alpha}\sum_t \boldsymbol{Z}_t\right) \geq b_3 - \epsilon \middle| X\right) \geq 1 - n\exp\left(\frac{-\alpha\epsilon^2}{8(b_2-b_1)^2}\right) \text{for all } X \in \mathcal{C}$$

Corollary 2.7.6 (Matrix Hoeffding conditioned on another random variable for an arbitrary nonzero mean matrix)**.** *Given an α-length sequence $\{\boldsymbol{Z}_t\}$ of random matrices of size $n_1 \times n_2$, a r.v. X, and a set $\mathcal{C}$ of values that X can take. Assume that, for all $X \in \mathcal{C}$, (i) $\boldsymbol{Z}_t$'s are conditionally independent given X; (ii) $\mathbb{P}(\|\boldsymbol{Z}_t\|_2 \leq b_1|X) = 1$ and (iii) $\|\frac{1}{\alpha}\sum_t \mathbb{E}(\boldsymbol{Z}_t|X)\|_2 \leq b_2$. Then, for all $\epsilon > 0$,*

$$\mathbb{P}\left(\left\|\frac{1}{\alpha}\sum_t \boldsymbol{Z}_t\right\|_2 \leq b_2 + \epsilon \middle| X\right) \geq 1 - (n_1+n_2)\exp\left(\frac{-\alpha\epsilon^2}{32{b_1}^2}\right) \text{ for all } X \in \mathcal{C}$$

2.7.3 Simple Lemmas Needed for the Proofs

Lemma 2.7.7. *For $j = 1, \ldots, J$ and $k = 1, \ldots, K$, for all $X_{\hat{u}_j+k-1} \in \Gamma_{j,k-1}^{\hat{u}_j}$*

1. $\mathbf{0} \preceq \mathbb{E}\left[\boldsymbol{a}_{t,*}{\boldsymbol{a}_{t,*}}' \;\middle|\; X_{\hat{u}_j+k-1}\right] = \boldsymbol{\Lambda}_{t,*} \preceq \lambda^+\boldsymbol{I}$

2. $\lambda_{\text{new}}^{-}\boldsymbol{I} \preceq \mathbb{E}\left[\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,\text{new}}{}' \;\middle|\; X_{\hat{u}_j+k-1}\right] = \boldsymbol{\Lambda}_{t,\text{new}} \preceq \lambda_{\text{new}}^{+}\boldsymbol{I}$ *and* $\hat{\lambda}_{\text{train}}^{-} \leq \lambda_{\text{new}}^{-} \leq \lambda_{\text{new}}^{+} \leq 3\hat{\lambda}_{\text{train}}^{-}$

3. $\mathbb{E}\left[\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,\text{new}}{}' \;\middle|\; X_{\hat{u}_j+k-1}\right] = \boldsymbol{0}$

with $\hat{u}_j = u_j$ *or* $\hat{u}_j = u_j + 1$.

The same bounds also hold for summation over $t \in \mathcal{J}_{u_j+1}$ *when we condition on* $X_{u_j} \in \Gamma_{j-1,\text{end}}$.

Proof. The proof follows from Model 2.2.2 and Fact 2.6.13. The only reason we need $X_{\hat{u}_j+k-1} \in \Gamma_{j,k-1}^{\hat{u}_j}$ is to apply Fact 2.6.13 which allows us to lower and upper bound in the eigenvalues of $\boldsymbol{\Lambda}_{t,\text{new}}$ by λ_{new}^{-} and λ_{new}^{+} and then use (3b). ⊠

Lemma 2.7.8. *Assume that the assumptions of Theorem 2.2.7 hold. Recall that* $\boldsymbol{D}_{\text{new}} = \boldsymbol{D}_{\text{new},0}$. *Conditioned on* $X_{k-1} \in \Gamma_{k-1}$,

$$\|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{D}_{\text{new}}\|_2 \leq \kappa_s^{+} := .0215 \tag{2.25}$$

for all $\mathcal{T}$ *such that* $|\mathcal{T}| \leq s$.

The proof is in Appendix 2.B.

2.7.4 Proofs of Lemma 2.6.21 and 2.6.22

Proof of Lemma 2.6.21. We obtain the bounds on $\boldsymbol{A}_u$ for $u = \hat{u}_j + k$ for $k = 1, 2, \ldots, K$ and $\hat{u}_j = u_j$ or $u_j + 1$. For $u = \hat{u}_j + k$, recall that $\boldsymbol{A}_u := \frac{1}{\alpha}\sum_t \boldsymbol{E}_{\text{new}}{}'\tilde{\boldsymbol{\ell}}_t\tilde{\boldsymbol{\ell}}_t'\boldsymbol{E}_{\text{new}}$.

Notice that $\boldsymbol{E}_{\text{new}}{}'\tilde{\boldsymbol{\ell}}_t = \boldsymbol{R}_{\text{new}}\boldsymbol{a}_{t,\text{new}} + \boldsymbol{E}_{\text{new}}{}'\boldsymbol{D}_*\boldsymbol{a}_{t,*}$. Let $\boldsymbol{Z}_t = \boldsymbol{R}_{\text{new}}\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,\text{new}}{}'\boldsymbol{R}_{\text{new}}{}'$, and let $\boldsymbol{Y}_t = \boldsymbol{R}_{\text{new}}\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,*}{}'\boldsymbol{D}_*{}'\boldsymbol{E}_{\text{new}} + \boldsymbol{E}_{\text{new}}{}'\boldsymbol{D}_*\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,\text{new}}{}'\boldsymbol{R}_{\text{new}}{}'$, then

$$\boldsymbol{A}_u \succeq \frac{1}{\alpha}\sum_t \boldsymbol{Z}_t + \frac{1}{\alpha}\sum_t \boldsymbol{Y}_t \tag{2.26}$$

Consider $\frac{1}{\alpha}\sum_t \boldsymbol{Z}_t$. (1) The $\boldsymbol{Z}_t$'s are conditionally independent given X_{k-1}. (2) With probability 1, $\|\boldsymbol{Z}_t\|_2 \leq r_{\text{new}}{\gamma_{\text{new}}}^2$. (3) Using a theorem of Ostrowoski [24, Theorem 4.5.9], conditioned on $X_{k-1} \in \Gamma_{k-1}$, $\lambda_{\min}\left(\mathbb{E}[\frac{1}{\alpha}\sum_t \boldsymbol{Z}_t | X_{k-1}]\right) = \lambda_{\min}\left(\boldsymbol{R}_{\text{new}}(\frac{1}{\alpha}\sum_t \boldsymbol{\Lambda}_{t,\text{new}})\boldsymbol{R}_{\text{new}}{}'\right) \geq \lambda_{\min}\left(\boldsymbol{R}_{\text{new}}\boldsymbol{R}_{\text{new}}{}'\right)\lambda_{\min}\left(\frac{1}{\alpha}\sum_t \boldsymbol{\Lambda}_{t,\text{new}}\right) \geq (1 - (\zeta_*^{+})^2)\lambda_{\text{new}}^{-}$. The last inequality uses Lemma 2.7.7 and Fact 2.7.4.

Thus, applying Corollary 2.7.5 with ϵ given by (2.20), we get that, for all $X_{k-1} \in \Gamma_{k-1}$,

$$\mathbb{P}\left(\lambda_{\min}\left(\frac{1}{\alpha}\sum_t \boldsymbol{Z}_t\right) \geq (1-(\zeta_*^+)^2)\lambda_{\text{new}}^- - \epsilon \middle| X_{k-1}\right) \geq 1 - r_{\text{new}} \exp\left(\frac{-\alpha\zeta^2(\hat{\lambda}_{\text{train}}^-)^2}{8\cdot 100^2 \cdot {\gamma_{\text{new}}}^4}\right). \quad (2.27)$$

Consider $\boldsymbol{Y}_t = \boldsymbol{R}_{\text{new}}\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,*}{}'\boldsymbol{D}_*{}'\boldsymbol{E}_{\text{new}} + \boldsymbol{E}_{\text{new}}{}'\boldsymbol{D}_*\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,\text{new}}{}'\boldsymbol{R}_{\text{new}}{}'$. (1) The $\boldsymbol{Y}_t$'s are conditionally independent given X_{k-1}. (2) Using the bound on ζ from the theorem, $\|\boldsymbol{Y}_t\| \leq 2\sqrt{r_{\text{new}}r}\zeta_*^+\gamma\gamma_{\text{new}} \leq 2\sqrt{r_{\text{new}}r}\zeta_*^+\gamma^2 \leq 2$ holds with probability one for all $X_{k-1} \in \Gamma_{k-1}$. Thus, under the same conditioning, $-2\boldsymbol{I} \preceq \boldsymbol{Y}_t \preceq 2\boldsymbol{I}$ with with probability one. (3) By Lemma 2.7.7, $\mathbb{E}\left(\frac{1}{\alpha}\sum_t \boldsymbol{Y}_t | X_{k-1}\right) = \boldsymbol{0}$ for all $X_{k-1} \in \Gamma_{k-1}$.

Thus, applying Corollary 2.7.5 with ϵ given by (2.20), we get that, for all $X_{k-1} \in \Gamma_{k-1}$

$$\mathbb{P}\left(\lambda_{\min}\left(\frac{1}{\alpha}\sum_t \boldsymbol{Y}_t\right) \geq -\epsilon \middle| X_{k-1}\right) \geq 1 - c\exp\left(\frac{-\alpha {r_{\text{new}}}^2\zeta^2(\hat{\lambda}_{\text{train}}^-)^2}{8\cdot 100^2\cdot(4)^2}\right) \quad (2.28)$$

Combining (2.26), (2.27) and (2.28) and using the union bound, we get the lemma. ☒

Proof of Lemma 2.6.22. Remark 2.7.3 applies.

We obtain the bounds on $\boldsymbol{A}_{u,\perp}$ for $u = \hat{u}_j + k$ for $k = 1, 2, ...K$ with $\hat{u}_j = u_j$ or $u_j + 1$. For all these u's, recall that $\boldsymbol{A}_{u,\perp} := \frac{1}{\alpha}\sum_t \boldsymbol{E}_{\text{new},\perp}{}'\tilde{\boldsymbol{\ell}}_t\tilde{\boldsymbol{\ell}}_t{}'\boldsymbol{E}_{\text{new},\perp}$. Using $\boldsymbol{E}_{\text{new},\perp}{}'\boldsymbol{D}_{\text{new}} = 0$, we get that $\boldsymbol{E}_{\text{new},\perp}{}'\tilde{\boldsymbol{\ell}}_t = \boldsymbol{E}_{\text{new},\perp}{}'\boldsymbol{D}_*\boldsymbol{a}_{t,*}$. Thus, $\boldsymbol{A}_{u,\perp} = \frac{1}{\alpha}\sum_t \boldsymbol{Z}_t$ with $\boldsymbol{Z}_t = \boldsymbol{E}_{\text{new},\perp}{}'\boldsymbol{D}_*\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,*}{}'\boldsymbol{D}_*{}'\boldsymbol{E}_{\text{new},\perp}$.

Using the same ideas as for the previous proof we can show that $\boldsymbol{0} \preceq \boldsymbol{Z}_t \preceq r(\zeta_*^+)^2\gamma^2\boldsymbol{I} \preceq \zeta\boldsymbol{I}$ and $\mathbb{E}\left(\frac{1}{\alpha}\sum_t \boldsymbol{Z}_t | X_{k-1}\right) \preceq (\zeta_*^+)^2\lambda^+\boldsymbol{I}$. Thus by Corollary 2.7.5 the lemma follows. ☒

2.7.5 Proof of Lemma 2.6.23

Proof of Lemma 2.6.23. Remark 2.7.3 applies. Using the expression for $\boldsymbol{\mathcal{H}}_u$ given in Definition 2.6.9, and noting that for a basis matrix $\boldsymbol{E}$, $\boldsymbol{E}\boldsymbol{E}' + \boldsymbol{E}_\perp\boldsymbol{E}_\perp{}' = \boldsymbol{I}$ we get that

$$\boldsymbol{\mathcal{H}}_u = \frac{1}{\alpha}\sum_{t\in\mathcal{J}_u}\Big((\boldsymbol{I}-\hat{\boldsymbol{P}}_*\hat{\boldsymbol{P}}_*{}')\boldsymbol{e}_t\boldsymbol{e}_t{}'(\boldsymbol{I}-\hat{\boldsymbol{P}}_*\hat{\boldsymbol{P}}_*{}') - (\tilde{\boldsymbol{\ell}}_t\boldsymbol{e}_t{}'(\boldsymbol{I}-\hat{\boldsymbol{P}}_*\hat{\boldsymbol{P}}_*{}') + (\boldsymbol{I}-\hat{\boldsymbol{P}}_*\hat{\boldsymbol{P}}_*{}')\boldsymbol{e}_t\tilde{\boldsymbol{\ell}}_t{}') + (\boldsymbol{F}_t + \boldsymbol{F}_t{}')\Big)$$

where

$$\boldsymbol{F}_t = \boldsymbol{E}_{\text{new},\perp}\boldsymbol{E}_{\text{new},\perp}{}'\tilde{\boldsymbol{\ell}}_t\tilde{\boldsymbol{\ell}}_t{}'\boldsymbol{E}_{\text{new}}\boldsymbol{E}_{\text{new}}{}'.$$

Thus,

$$\|\boldsymbol{\mathcal{H}}_u\|_2 \leq 2\left\|\frac{1}{\alpha}\sum_t \tilde{\boldsymbol{\ell}}_t\boldsymbol{e}_t{}'\right\|_2 + \left\|\frac{1}{\alpha}\sum_t \boldsymbol{e}_t\boldsymbol{e}_t{}'\right\|_2 + 2\left\|\frac{1}{\alpha}\sum_t \boldsymbol{F}_t\right\|_2 \quad (2.29)$$

Next we obtain high probability bounds on each of the three terms on the right hand side of (2.29).

Consider $\left\|\frac{1}{\alpha}\sum_t \tilde{\boldsymbol{\ell}}_t \boldsymbol{e}_t{}'\right\|_2$. Using Lemma 2.6.15, $\boldsymbol{e}_t$ satisfies (2.18) with probability one for all $X_{k-1} \in \Gamma_{k-1}$.

Let $\boldsymbol{Z}_t := \tilde{\boldsymbol{\ell}}_t \boldsymbol{e}_t{}'$. (1) Conditioned on X_{k-1}, the various $\boldsymbol{Z}_t$'s used in the summation are mutually independent, for all $X_{k-1} \in \Gamma_{k-1}$. (2) For $X_{k-1} \in \Gamma_{k-1}$,

$$\|\boldsymbol{Z}_t\|_2 = \|\tilde{\boldsymbol{\ell}}_t \boldsymbol{e}_t{}'\|_2 \le \left(\zeta_*^+\sqrt{r}\gamma + \sqrt{r_{\text{new}}}\gamma_{\text{new}}\right)\left(\phi^+(\zeta_*^+\sqrt{r}\gamma + \zeta_{\text{new},k-1}^+\sqrt{r_{\text{new}}}\gamma_{\text{new}})\right) := b_3$$

holds with probability one. (3) First consider the $k \ge 2$ case. When $X_{k-1} \in \Gamma_{k-1}$,

$$\begin{aligned}
&\left\|\mathbb{E}\left[\frac{1}{\alpha}\sum_t \tilde{\boldsymbol{\ell}}_t \boldsymbol{e}_t{}' \mid X_{k-1}\right]\right\|_2 \\
=&\left\|\frac{1}{\alpha}\sum_t \left[\left(\boldsymbol{D}_*\boldsymbol{\Lambda}_{t,*}\boldsymbol{D}_{*,k-1}{}' + \boldsymbol{D}_{\text{new}}\boldsymbol{\Lambda}_{t,\text{new}}\boldsymbol{D}_{\text{new},k-1}{}'\right)\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\right]\right\|_2 \\
\le&\Bigg[\lambda_{\max}\Bigg(\frac{1}{\alpha}\sum_t \left(\boldsymbol{D}_*\boldsymbol{\Lambda}_{t,*}\boldsymbol{D}_{*,k-1}{}' + \boldsymbol{D}_{\text{new}}\boldsymbol{\Lambda}_{t,\text{new}}\boldsymbol{D}_{\text{new},k-1}{}'\right) \\
&\qquad\qquad \left(\boldsymbol{D}_*\boldsymbol{\Lambda}_{t,*}\boldsymbol{D}_{*,k-1}{}' + \boldsymbol{D}_{\text{new}}\boldsymbol{\Lambda}_{t,\text{new}}\boldsymbol{D}_{\text{new},k-1}{}'\right)'\Bigg)\Bigg]^{1/2} \\
&\sqrt{\lambda_{\max}\left(\frac{1}{\alpha}\sum_t \left(\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\right)\left(\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\right)'\right)} \\
\le&\left((\zeta_*^+)^2\lambda^+ + \zeta_{\text{new},k-1}^+\lambda_{\text{new}}^+\right)\left(\sqrt{\rho^2 h^+}\phi^+\right).
\end{aligned}$$

The first inequality is by Cauchy-Schwarz for a sum of matrices. This can be found as Lemma 2.D.2 in Appendix 2.D. The second inequality uses Fact 2.7.4 (for the first term of the product) and Lemma 2.5.3 with $\sigma^+ = (\phi^+)^2$ (for the second term of the product).

Now consider the $k = 1$ case. To bound $\left\|\frac{1}{\alpha}\sum_t \boldsymbol{D}_*\boldsymbol{\Lambda}_{t,*}\boldsymbol{D}_{*,0}{}'\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\right\|_2$ we proceed exactly as we did for the $k \ge 2$ case. We can bound this by $(\zeta_*^+)^2\lambda^+\sqrt{\rho^2 h^+}\phi^+$. To bound $\left\|\frac{1}{\alpha}\sum_t \boldsymbol{D}_{\text{new}}\boldsymbol{\Lambda}_{t,\text{new}}\boldsymbol{D}_{\text{new},0}{}'\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\right\|_2$, we apply Lemma 2.7.8 to get[7] $\|\boldsymbol{D}_{\text{new},0}{}'\boldsymbol{I}_{\mathcal{T}_t}\|_2 \le \kappa_s^+$. Using this and Fact 2.7.4, we can bound this by $\kappa_s^+\lambda_{\text{new}}^+\phi^+$. Thus, when

[7]Notice that if we want to use the bound of Lemma 2.7.8, we cannot also apply Lemma 2.5.3 for this term. We can get a simpler proof by not using Lemma 2.7.8 at all and proceeding exactly as we did for the $k \ge 2$ case; but doing this will require a much tighter bound on $\rho^2 h^+$ than what we currently need.

$X_0 \in \Gamma_0$,

$$\begin{aligned}
&\left\| \mathbb{E}\left[\frac{1}{\alpha} \sum_t \tilde{\boldsymbol{\ell}}_t \boldsymbol{e}_t{}' \mid X_0 \right] \right\|_2 \\
=&\left\| \frac{1}{\alpha} \sum_t \left[\left(\boldsymbol{D}_* \boldsymbol{\Lambda}_{t,*} \boldsymbol{D}_*{}' + \boldsymbol{D}_{\text{new}} \boldsymbol{\Lambda}_{t,\text{new}} \boldsymbol{D}_{\text{new}}{}' \right) \boldsymbol{I}_{\mathcal{T}_t} [(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}' (\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1} \boldsymbol{I}_{\mathcal{T}_t}{}' \right] \right\|_2 \\
\leq& \left(\sqrt{\rho^2 h^+} (\zeta_*^+)^2 \lambda^+ + \kappa_s^+ \lambda_{\text{new}}^+ \right) \phi^+.
\end{aligned}$$

Thus, by Corollary 2.7.6 with ϵ given by (2.20), we get that, for all $X_{k-1} \in \Gamma_{k-1}$,

$$\mathbb{P}\left(\left\| \frac{1}{\alpha} \sum_t \tilde{\boldsymbol{\ell}}_t \boldsymbol{e}_t{}' \right\|_2 \leq b_{\boldsymbol{\ell}\boldsymbol{e},k} \middle| X_{k-1} \right) \geq 1 - n \exp\left(\frac{-\alpha {r_{\text{new}}}^2 \zeta^2 (\hat{\lambda}_{\text{train}}^-)^2}{32 \cdot 100^2 {b_3}^2} \right). \tag{2.30}$$

Consider $\| \frac{1}{\alpha} \sum_t \boldsymbol{e}_t \boldsymbol{e}_t{}' \|_2$. Let $\boldsymbol{Z}_t = \boldsymbol{e}_t \boldsymbol{e}_t{}'$. (1) Conditioned on X_{k-1}, the various $\boldsymbol{Z}_t$'s in the summation are independent, for all $X_{k-1} \in \Gamma_{k-1}$. (2) Using Lemma 2.6.15, conditioned on $X_{k-1} \in \Gamma_{k-1}$,

$$\boldsymbol{0} \preceq \boldsymbol{Z}_t \preceq \left(\phi^+ (\zeta_*^+ \sqrt{r} \gamma + \zeta_{\text{new},k-1}^+ \sqrt{r_{\text{new}}} \gamma_{\text{new}}) \right)^2 \boldsymbol{I} := b_1 \boldsymbol{I}$$

with probability one. (3) By Fact 2.7.4, when $X_{k-1} \in \Gamma_{k-1}$,

$$\begin{aligned}
&\frac{1}{\alpha} \sum_t \mathbb{E}\left[\boldsymbol{e}_t \boldsymbol{e}_t{}' | X_{k-1} \right] \\
&= \frac{1}{\alpha} \sum_t \boldsymbol{I}_{\mathcal{T}_t} [(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}' (\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1} \boldsymbol{I}_{\mathcal{T}_t}{}' \Big(\boldsymbol{D}_{*,k-1} \boldsymbol{\Lambda}_{t,*} \boldsymbol{D}_{*,k-1}{}' + \\
&\qquad\qquad \boldsymbol{D}_{\text{new},k-1} \boldsymbol{\Lambda}_{t,\text{new}} \boldsymbol{D}_{\text{new},k-1}{}' \Big) \boldsymbol{I}_{\mathcal{T}_t} [(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}' (\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1} \boldsymbol{I}_{\mathcal{T}_t}{}'
\end{aligned}$$

When $k = 1$ we can apply Lemma 2.7.8 to get that $\| \boldsymbol{D}_{\text{new},0}{}' \boldsymbol{I}_{\mathcal{T}_t} \|_2 \leq \kappa_s^+$. Then we apply Lemma 2.5.3 with $\sigma^+ = (\phi^+)^2 \left((\zeta_*^+)^2 \lambda^+ + (\kappa_s^+)^2 \lambda_{\text{new}}^+ \right)$. This gives

$$\boldsymbol{0} \preceq \mathbb{E}\left[\sum_t \boldsymbol{e}_t \boldsymbol{e}_t{}' \middle| X_0 \right] \preceq \rho^2 h^+ (\phi^+)^2 \Big((\zeta_*^+)^2 \lambda^+ + (\kappa_s^+)^2 \lambda_{\text{new}}^+ \Big) \boldsymbol{I} \quad \text{for all } X_0 \in \Gamma_0.$$

When $k \geq 2$ we can apply Lemma 2.5.3 with $\sigma^+ = (\phi^+)^2 \left((\zeta_*^+)^2 \lambda^+ + (\zeta_{\text{new},k-1}^+)^2 \lambda_{\text{new}}^+ \right)$ to get that,

$$\boldsymbol{0} \preceq \mathbb{E}\left[\sum_t \boldsymbol{e}_t \boldsymbol{e}_t{}' \middle| X_{k-1} \right] \preceq \rho^2 h^+ (\phi^+)^2 \Big((\zeta_*^+)^2 \lambda^+ + (\zeta_{\text{new},k-1}^+)^2 \lambda_{\text{new}}^+ \Big) \boldsymbol{I} \quad \text{for all } X_{k-1} \in \Gamma_{k-1}.$$

Thus, applying Corollary 2.7.5 with ϵ given by (2.20), we get that, for all $X_{k-1} \in \Gamma_{k-1}$,

$$\mathbb{P}\left(\left\| \frac{1}{\alpha} \sum_t \boldsymbol{e}_t \boldsymbol{e}_t{}' \right\|_2 \leq b_{\boldsymbol{e}\boldsymbol{e},k} \middle| X_{k-1} \right) \geq 1 - n \exp\left(\frac{-\alpha r_{\text{new}}^2 \zeta^2 (\hat{\lambda}_{\text{train}}^-)^2}{8 \cdot 100^2 {b_1}^2} \right). \tag{2.31}$$

Finally, consider $\left\|\frac{1}{\alpha}\sum_t \boldsymbol{F}_t\right\|_2$. Since $\boldsymbol{E}_{\text{new},\perp}{}'\boldsymbol{D}_{\text{new}} = 0$,

$$\begin{aligned}\boldsymbol{F}_t &= \boldsymbol{E}_{\text{new},\perp}\boldsymbol{E}_{\text{new},\perp}{}'\tilde{\boldsymbol{\ell}}_t\tilde{\boldsymbol{\ell}}_t{}'\boldsymbol{E}_{\text{new}}\boldsymbol{E}_{\text{new}}{}' \\ &= \boldsymbol{E}_{\text{new},\perp}\boldsymbol{E}_{\text{new},\perp}{}'(\boldsymbol{D}_*\boldsymbol{a}_{t,*})(\boldsymbol{D}_*\boldsymbol{a}_{t,*} + \boldsymbol{D}_{\text{new}}\boldsymbol{a}_{t,\text{new}})'\boldsymbol{E}_{\text{new}}\boldsymbol{E}_{\text{new}}{}'\end{aligned}$$

(1) Conditioned on X_{k-1}, the $\boldsymbol{F}_t$'s are mutually independent, for all $X_{k-1} \in \Gamma_{k-1}$. (2) For $X_{k-1} \in \Gamma_{k-1}$,

$$\|\boldsymbol{F}_t\|_2 \leq (\zeta_*^+)^2 r\gamma^2 + \zeta_*^+\sqrt{r r_{\text{new}}}\gamma\gamma_{\text{new}} := b_5$$

holds with probability 1. (3) For $X_{k-1} \in \Gamma_{k-1}$,

$$\left\|\mathbb{E}\Big[\frac{1}{\alpha}\sum_t \boldsymbol{F}_t \mid X_{k-1}\Big]\right\|_2 \leq \left\|\frac{1}{\alpha}\sum_t(\boldsymbol{D}_*\boldsymbol{\Lambda}_{t,*}\boldsymbol{D}_*{}')\right\|_2 \leq (\zeta_*^+)^2\lambda^+ = b_{\boldsymbol{F}}$$

Applying Corollary 2.7.6 with ϵ given by (2.20), we get that, for all $X_{k-1} \in \Gamma_{k-1}$,

$$\mathbb{P}\left(\left\|\frac{1}{\alpha}\sum_t \boldsymbol{F}_t\right\|_2 \leq b_{\boldsymbol{F}}\middle| X_{k-1}\right) \geq 1 - n\exp\left(\frac{-\alpha {r_{\text{new}}}^2\zeta^2(\hat{\lambda}_{\text{train}}^-)^2}{32\cdot 100^2 {b_5}^2}\right) \tag{2.32}$$

Combining (2.29) with (2.30), (2.31) and (2.32) and using the union bound, we get the lemma. The expression for $p_{\mathcal{H}}$ given in the lemma uses the bounds on ζ from the theorem and uses the loose bound $\zeta_{j,\text{new},k-1}^+ \leq 1$ (to get a simpler expression for the probabilities). ⊠

2.8 Simulation Experiments

In this section we provide some simulations that demonstrate the robust PCA result we have proven above. More detailed simulations using real data can be found in [17].

The data for Figure 2.7 was generated as follows. We chose $n = 256$ and $t_{\max} = 15,000$. Each measurement had $s = 20$ missing or corrupted entries, i.e. $|\mathcal{T}_t| = 20$. Each non-zero entry of $\boldsymbol{x}_t$ was drawn uniformly at random between 2 and 6 independent of other entries and other times t. In Figure 2.7 the support of $\boldsymbol{x}_t$ changes as assumed in Model 2.2.3 with $\rho = 2$ and $\beta = 18$. So the support of $\boldsymbol{x}_t$ changes by $\frac{s}{2} = 10$ indices every 18 time instants. When the support of $\boldsymbol{x}_t$ reaches the bottom of the vector, it starts over again at the top. This pattern can be seen in the bottom half of the figure which shows the sparsity pattern of the matrix $\boldsymbol{S} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{t_{\max}}]$.

To form the low dimensional vectors $\boldsymbol{\ell}_t$, we started with an $n \times r$ matrix of i.i.d. Gaussian entries and orthonormalized the columns. The first $r_0 = 10$ columns of this matrix formed $\boldsymbol{P}_{(0)}$, the next 2 columns formed $\boldsymbol{P}_{(1),\text{new}}$, and the last 2 columns formed $\boldsymbol{P}_{(2),\text{new}}$ We show two subspace changes which occur at $t_1 = 600$ and $t_2 = 8000$. The entries of $\boldsymbol{a}_{t,*}$ were drawn uniformly at random between -5 and 5, and the entries of $\boldsymbol{a}_{t,\text{new}}$ were drawn uniformly at random between $-\sqrt{3v_i^{t-t_j}\hat{\lambda}_{\text{train}}^-}$ and $\sqrt{3v_i^{t-t_j}\hat{\lambda}_{\text{train}}^-}$ with $v_i = 1.00017$ and $\hat{\lambda}_{\text{train}}^- = 1$ (and $q_i = 1$). Thus $(\boldsymbol{\Lambda}_{t,\text{new}})_{i,i} = v_i^{t-t_j}\hat{\lambda}_{\text{train}}^-$ as assumed in Model 2.2.2. Entries of $\boldsymbol{a}_t$ were independent of each other and of the other $\boldsymbol{a}_t$'s.

For this simulated data we compare the performance of ReProCS and PCP. The plots show the relative error in recovering $\boldsymbol{\ell}_t$, that is $\|\boldsymbol{\ell}_t - \hat{\boldsymbol{\ell}}_t\|_2/\|\boldsymbol{\ell}_t\|_2$. For the initial subspace estimate $\hat{\boldsymbol{P}}_0$, we used $\boldsymbol{P}_0$ plus some small Gaussian noise and then obtained orthonormal columns. We set $\alpha = 800$ and $K = 6$. For the PCP algorithm, we perform the optimization every α time instants using all of the data up to that point. So the first time PCP is performed on $[\boldsymbol{m}_1, \dots, \boldsymbol{m}_\alpha]$ and the second time it is performed on $[\boldsymbol{m}_1, \dots, \boldsymbol{m}_{2\alpha}]$ and so on.

Figure 2.7 illustrates the result we have proven. That is ReProCS takes advantage of the initial subspace estimate and slow subspace change (including the bound on γ_{new}) to handle the case when the supports of $\boldsymbol{x}_t$ are correlated in time. Notice how the ReProCS error increases after a subspace change, but decays exponentially with each projection PCA step. For this data, the PCP program fails to give a meaningful estimate for all but a few times. The average time taken by the ReProCS algorithm was 52 seconds, while PCP averaged over 5 minutes. Simulations were coded in MATLAB® and run on a desktop computer with a 3.2 GHz processor.

2.9 Extensions

In this section, we first give other models on changes in $\mathcal{T}_t$ that are special cases of the general model Model 2.5.1 and hence can also be used in Theorem 2.2.5 or 2.2.7. The next three subsections discuss various other results that can also be proved using the proof techniques developed in this work.

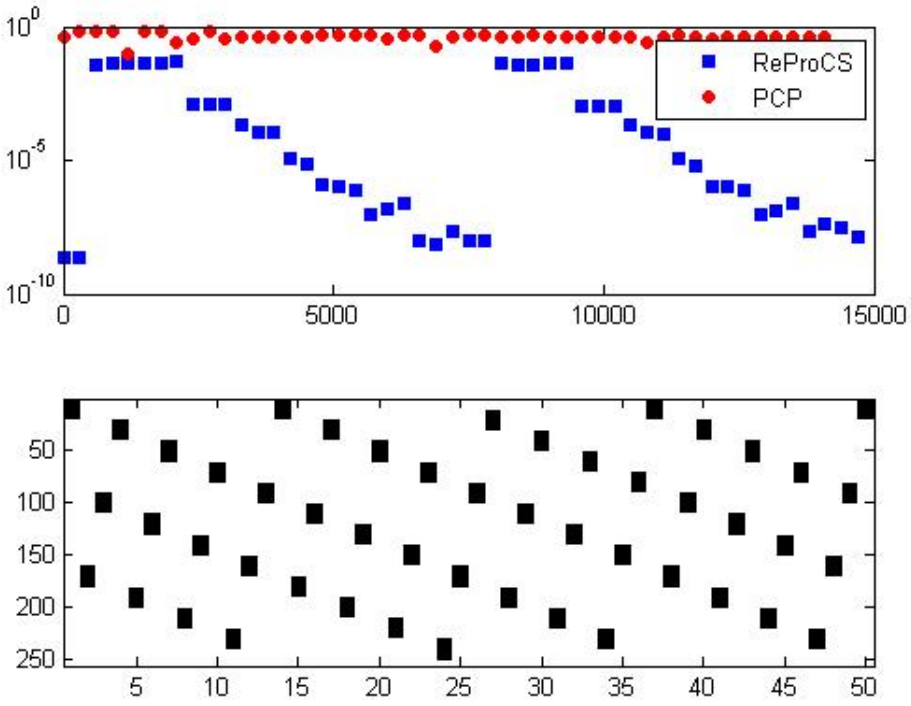

Figure 2.7: Comparison of ReProCS and PCP for the RPCA problem. The top plot is the relative error $\|\boldsymbol{\ell}_t - \hat{\boldsymbol{\ell}}_t\|_2 / \|\boldsymbol{\ell}_t\|_2$. The bottom plot shows the sparsity pattern of $\boldsymbol{S}$ (black represents a non-zero entry). Results are averaged over 100 simulations and plotted every 300 time instants.

2.9.1 Other Models on Changes in $\mathcal{T}_t$

We give here other models on changes in $\mathcal{T}_t$ that are special cases of Model 2.5.1.

Model 2.9.1. *Suppose that $\mathcal{T}_t$ consists of consecutive indices and is of size s or less, i.e. $|\mathcal{T}_t| \leq s$. When $\mathcal{T}_t$ is not empty, let $\tilde{o}_t$ denote its smallest (topmost) index. Let ρ_1 be an integer. We assume that $\tilde{o}_t$ satisfies the following Bernoulli-Gaussian model:*

$$\tilde{o}_t = \lceil o_t \mod n \rceil \text{ where } o_t = o_{t-1} + \theta_t \left(1.1 \frac{s}{\rho} + \varpi_t \right)$$

where $\varpi_t \sim \mathcal{N}(0, \sigma^2)$ (Gaussian) and $\theta_t \sim$ Bernoulli(q). Assume that $\{\varpi_t\}$, $\{\theta_t\}$ are mutually independent and independent of $\boldsymbol{\ell}_t$'s. Taking the mod with respect to n describes the process of the set $\mathcal{T}_t$ starting over at 1 when its topmost index exceeds n (this models a new object appearing after the old one has disappeared; notice that at any t $\mathcal{T}_t$ could be empty as well, i.e.

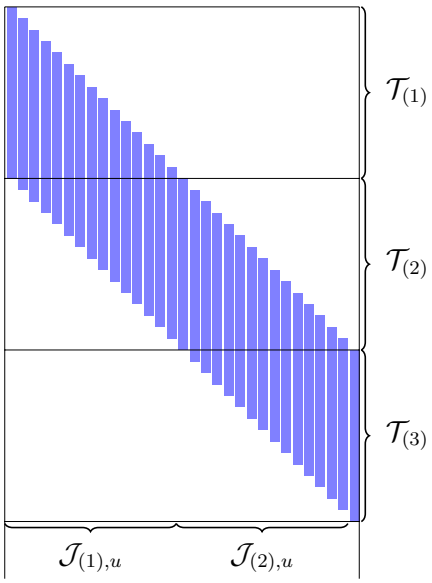

Figure 2.8: Model 2.9.2

there may be no object).

Assume that $s \leq \frac{1.2\rho n}{\alpha}$, $q \geq 1 - (\frac{n^{-10}}{2t_{\max}})^{\frac{1}{\beta}}$ *for a* β *that satisfies* $\rho^2 \frac{\beta}{\alpha} \leq 0.01$, *and* $\sigma^2 \leq \frac{s^2}{4000\rho^2 \log(n)}$.

Model 2.9.2. *Suppose that* $\mathcal{T}_t$ *consists of* s *consecutive indices and suppose that it moves down the vector by between 1 and* m *indices at every time* t. *When it reaches the bottom of the vector, we assume that it starts over at* 1. *Assume that* $s \leq 0.0025\alpha$ *and* $m \leq \frac{n-s}{\alpha}$.

Model 2.9.3. *In both models above we let* $\mathcal{T}_t$ *contain consecutive indices. This models a moving 1D object of length* s *or less that enters the scene and eventually walks out, and then another object of length* s *or less may come in. However notice that nothing in our general model, Model 2.5.1, requires the indices to be consecutive or contiguous in any way. Thus in both of Models 2.9.1 and 2.9.2 above, instead of one moving object, we can also have multiple moving objects as long as the union of their supports is of size at most* s *and satisfies one of these models. Also, with minor changes, the object(s) instead of leaving the scene can reflect back up and start moving in the other direction as well.*

Lemma 2.9.4. *If* $t_{\max} \leq n^{10}$, *then Model 2.9.1 is a special case of Model 2.2.3 (and hence a special case of Model 2.5.1) with probability at least* $1 - n^{-10}$.

Proof. The proof has three steps. (a) We first use standard arguments about a Bernoulli sequence [25] to prove that the object moves at least once every β time instants with probability at least $1-0.5n^{-10}$. The choice of q ensures that this holds. (b) Next we use a standard Gaussian tail bound argument to show that, with probability at least $1-0.5n^{-10}$, when it moves, it moves by at least s/ρ indices and at most $1.2s/\rho$ indices. The bound on σ^2 ensures this. (c) The above two claims ensure that, w.h.p., the object remains static for at most β frames at a time and when it moves it moves by at least s/ρ indices and at most $1.2s/\rho$ indices. Notice that all the motion is in one direction. Motion by at least s/ρ in one direction ensures that after the object moves ρ times, i.e. after ρ changes of $\mathcal{T}_t$, the sets are disjoint, i.e. $\mathcal{T}^{[k]} \cap \mathcal{T}^{[k+\rho]} = \emptyset$. Motion by at most $1.2s/\rho$ in one direction and $1.2\frac{s}{\rho}\alpha \leq n$ ensures the third condition of Model 2.2.3 holds even when the object moves at every frame. ⊠

Lemma 2.9.5. *Model 2.9.2 is a special case of Model 2.5.1 with $\rho = 2$ and $h^+ = s/\alpha$.*

See Figure 2.8 for a diagram of the model and the idea behind its proof.

Proof. For the sake of clarity, we will prove the case when the object moves exactly 1 index at every time t. The only difference in the general case is the construction of the $\mathcal{J}_{(i),u}$.

Consider an interval $\mathcal{J}_u$. Let $t_u := (u-1)\alpha + 1$ denote the first time in $\mathcal{J}_u$. Without loss of generality (because we can re-label the indices) let the object start at the top of the vector. That is $\mathcal{T}_{t_u} = [1, s]$. Let $l_u = \left\lceil \frac{n}{s} \right\rceil$. Let $\mathcal{T}_{(i),u} = [(i-1)s + 1, is]$ for $i = 1, 2, \ldots, \left\lfloor \frac{n}{s} \right\rfloor$. If $\frac{n}{s}$ is not an integer, also define $\mathcal{T}_{\left(\left\lceil \frac{n}{s} \right\rceil\right),u} = \left[\left\lfloor \frac{n}{s} \right\rfloor s + 1, n\right]$. Define $\mathcal{J}_{(i),u} := [t_u + (i-1)s, t_u + is - 1]$ for $i = 1, 2, \ldots, \left\lfloor \frac{\alpha}{s} \right\rfloor$. If $\frac{\alpha}{s}$ is not an integer, also define $\mathcal{J}_{\left(\left\lceil \frac{\alpha}{s} \right\rceil\right),u} = [t_u + \left\lfloor \frac{\alpha}{s} \right\rfloor s, t_u + \alpha - 1]$.

Clearly $\mathcal{J}_{(i),u}$ as defined above are a partition of $\mathcal{J}_u$. Also, by construction, for all $t \in \mathcal{J}_{(i),u}$, $\mathcal{T}_t \subseteq \mathcal{T}_{(i),u} \cup \mathcal{T}_{(i+1),u}$. This follows from three facts 1) the assumption that $\mathcal{T}_{t_u} = [1, s]$ (which is just a renumbering of the indices to make the numbers clearer) 2) the object moves down by exactly one index at each time t and 3) $m \leq \frac{n-s}{\alpha}$, so that once an index leaves $\mathcal{T}_t$, it will not return in the next α time instants. A simpler way of stating fact 3) is that the total motion is such that $\mathcal{T}_t$ does not return to where it started i.e. $\mathcal{T}_{t_u} \cap \mathcal{T}_{t_u+\alpha} = \emptyset$.

Notice that $|\mathcal{J}_{(i),u}| \leq s$ for all i. (With the possible exception of the last set, they all have size exactly s.) So under the assumptions of Model 2.9.2 $h_u^*(\alpha) \leq s$, which satisfies Model 2.5.1 with $h^+ = \frac{s}{\alpha} \leq 0.0025\alpha = \frac{0.01\alpha}{2^2} = \frac{0.01\alpha}{\rho^2}$. ⊠

2.9.2 Analyze the ReProCS algorithm that also removes the deleted directions from the subspace estimate

The tools introduced in this paper – (a) Lemma 2.5.3 and the way it is applied to bound $\boldsymbol{\mathcal{H}}_u$ in Lemma 2.6.23; and (b) the detection lemma (Lemma 2.6.17), the no false detection lemma (Lemma 2.6.16) and the p-PCA lemma (Lemma 2.6.18) – can also be used to get a correctness result for a practical modification of ReProCS with cluster-PCA (ReProCS-cPCA) which is Algorithm 2 of [11]. This algorithm was introduced to also remove the deleted directions from the subspace estimate. It does this by re-estimating the previous subspace at a time after the newly added subspace has been accurately estimated (i.e. at a time after $\hat{t}_j + K\alpha$). A partial result for this algorithm was proved in [11].

This result will need one extra assumption – it will need the eigenvalues of the covariance matrix of $\boldsymbol{\ell}_t$ to be clustered for a period of time after the subspace change has stabilized, i.e. for a period of d_2 frames in the interval $[t_j + d + 1, t_{j+1} - 1]$ – but it will have a key advantage. It will need a much weaker denseness assumption and hence a much weaker bound on r or r_{mat}. In particular, with this result we expect to be able to allow $r = r_{\text{mat}} \in \mathcal{O}(n)$ with the same assumptions on s and s_{mat} that we currently allow. This requirement is almost as weak as that of PCP.

2.9.3 Relax the independence assumption on ℓ_t's

The results in this work assume that the $\boldsymbol{\ell}_t$'s are independent over time and zero mean; this is a valid model when background images have independent random variations about a fixed mean. Using the tools developed in this paper, a similar result can also be obtained for the more general case of $\boldsymbol{\ell}_t$'s following an autoregressive model. This will allow the $\boldsymbol{\ell}_t$'s to be correlated over time. A partial result for this case was obtained in [26]. The main change in this case will be that we will need to apply the matrix Azuma inequality from [23] instead

of matrix Hoeffding. This is will also require algebraic manipulation of sums and some other important modifications, as explained in [26], so that the constant term after conditioning on past values of the matrix is small.

2.9.4 Noisy and Undersampled Online Matrix Completion or Online Robust PCA

We expect that the tools introduced in this paper can also be used to analyze the noisy case, i.e. the case of $\boldsymbol{m}_t = \boldsymbol{x}_t + \boldsymbol{\ell}_t + \boldsymbol{w}_t$ where $\boldsymbol{w}_t$ is small bounded noise. In most practical video applications, while the foreground is truly sparse, the background is only approximately low-rank. The modeling error can be handled as $\boldsymbol{w}_t$. The proposed algorithms already apply without modification to this case (see [17] for results on real videos). The reason that our tools will directly extend to the noisy case is this: the sparse recovery step is already a noisy sparse recovery one, its analysis will not change if we also add in more noise due to $\boldsymbol{w}_t$. If $\boldsymbol{\ell}_t$ and $\boldsymbol{w}_t$ are assumed independent, then there should be few simple modifications to the analysis of the p-PCA step as well.

Finally, we expect both the algorithm and the proof techniques to apply with simple changes to the undersampled case $\boldsymbol{m}_t = \boldsymbol{A}_t\boldsymbol{x}_t + \boldsymbol{B}_t\boldsymbol{\ell}_t + \boldsymbol{w}_t$ as long as $\boldsymbol{B}_t$ is *not* time-varying, i.e. $\boldsymbol{B}_t = \boldsymbol{B}_0$. A partial result for this case was obtained in [27] and experiments were shown in [17].

2.10 Conclusions

In this work, we obtained correctness results for online robust PCA and for online matrix completion. Both results needed four key assumptions: (a) accurate initial subspace knowledge; (b) slow subspace change and mutual independence of the $\boldsymbol{\ell}_t$'s according to Model 2.2.2; (c) *some* changes in the set of missing entries (or in the set of outlier-corrupted entries) over time, one way to quantify what is needed is given in Model 2.2.3; (d) a denseness assumption on the columns of the subspace basis matrices of $\boldsymbol{\ell}_t$; and (e) algorithm parameters are appropriately set.

Ongoing work includes obtaining the results mentioned in Sections 2.9.2, 2.9.3 and 2.9.4. Besides these, we expect the proof techniques developed here to apply to various other problems involving PCA with data and noise terms being correlated.

2.A Appendix A:

Proof that Model 2.2.3 on $\mathcal{T}_t$ satisfies the general Model 2.5.1

Proof of Lemma 2.5.2. Consider an interval $\mathcal{J}_u$. We will construct one set of mutually disjoints sets $\{\mathcal{T}_{(i),u}\}_{i=1,2,\dots l_u}$ that are subsets of $\{1,2,\dots n\}$ and a partition $\{\mathcal{J}_{(i),u}\}_{i=1,2,\dots l_u}$ of $\mathcal{J}_u$ so that for all $t \in \mathcal{J}_{(i),u}$, (2.10) holds and so that $h_u(\alpha;\{\mathcal{T}_{(i),u}\},\{\mathcal{J}_{(i),u}\}) \leq \beta$ for this choice. Since $h_u^*(\alpha)$ takes the minimum over all such sets, this will imply $h_u^*(\alpha) \leq \beta$. By setting $h^+ = \beta/\alpha$ and using the Model 2.2.3 assumption $\rho^2\beta \leq 0.01\alpha$, we will be done.

Recall from Model 2.2.3 that $\mathcal{T}_t = \mathcal{T}^{[k]}$ for all $t \in [t^k, t^{k+1})$ with $t^{k+1} - t^k < \beta$ and $|\mathcal{T}^{[k]}| \leq s$.

Let $t_u := (u-1)\alpha + 1$ denote the first time index of $\mathcal{J}_u$. Let k_u be the index k for which $t_u \in [t^k, t^{k+1})$. In other words, $\mathcal{T}_{t_u} = \mathcal{T}^{[k_u]}$. Define l_u to be the number of intervals $[t^k, t^{k+1})$ that have non-empty intersection with $\mathcal{J}_u$. So l_u is one plus the number of times $\mathcal{T}_t$ changes in the interval $\mathcal{J}_u$. For $i = 1, 2, \dots l_u - 1$, define

$$\mathcal{T}_{(i),u} := \mathcal{T}^{[k_u+i-1]} \setminus \mathcal{T}^{[k_u+i]},$$

and set $\mathcal{T}_{(l_u),u} = \mathcal{T}^{[k_u+l_u-1]}$. Clearly $l_u \leq \alpha$. Thus, by the Model 2.2.3 assumption (for any k and i such that $k < i \leq k+\alpha$, $(\mathcal{T}^{[k]} \setminus \mathcal{T}^{[k+1]}) \cap (\mathcal{T}^{[i]} \setminus \mathcal{T}^{[i+1]}) = \emptyset$), the $\mathcal{T}_{(i),u}$'s are mutually disjoint.

Next, define a partition of $\mathcal{J}_u$ as

$$\mathcal{J}_{(i),u} := [t^{k_u+i-1}, t^{k_u+i}) \cap \mathcal{J}_u \text{ for } i = 1, 2, \dots l_u$$

By Model 2.2.3 $1 \leq t_{k+1} - t_k < \beta$ for all k. Since $\mathcal{J}_{(i),u} \subseteq [t^{k_u+i-1}, t^{k_u+i})$, $|\mathcal{J}_{(i),u}| < \beta$ for all $i = 1, 2, \dots l_u$.

Notice that for all $t \in \mathcal{J}_{(i),u}$, $\mathcal{T}_t = \mathcal{T}^{[k_u+i-1]}$. So if we can show that $\mathcal{T}^{[k_u+i-1]} \subseteq \mathcal{T}_{(i),u} \cup \mathcal{T}_{(i+1),u} \cdots \cup \mathcal{T}_{(i+\rho-1),u}$ for all $i = 1, 2, \dots l_u$, we will be done since this will imply $h_u^*(\alpha) \leq \beta$.

To show this, set $k = k_u + i - 1$. Then,

$$\begin{aligned}
\mathcal{T}^{[k]} &= \mathcal{T}_{(i),u} \cup [\mathcal{T}^{[k]} \cap \mathcal{T}^{[k+1]}] \\
&= \mathcal{T}_{(i),u} \cup [\mathcal{T}^{[k]} \cap \mathcal{T}^{[k+1]} \setminus \mathcal{T}^{[k+2]}] \cup [\mathcal{T}^{[k]} \cap \mathcal{T}^{[k+1]} \cap \mathcal{T}^{[k+2]}] \\
&\subseteq \mathcal{T}_{(i),u} \cup \mathcal{T}_{(i+1),u} \cup [\mathcal{T}^{[k]} \cap \mathcal{T}^{[k+1]} \cap \mathcal{T}^{[k+2]}] \\
&= \mathcal{T}_{(i),u} \cup \mathcal{T}_{(i+1),u} \cup [\mathcal{T}^{[k]} \cap \mathcal{T}^{[k+1]} \cap \mathcal{T}^{[k+2]} \setminus \mathcal{T}^{[k+3]}) \cup [\mathcal{T}^{[k]} \cap \mathcal{T}^{[k+1]} \cap \mathcal{T}^{[k+2]} \cap \mathcal{T}^{[k+3]}] \\
&\subseteq \mathcal{T}_{(i),u} \cup \mathcal{T}_{(i+1),u} \cup \mathcal{T}_{(i+2)} \cup [\mathcal{T}^{[k]} \cap \mathcal{T}^{[k+1]} \cap \mathcal{T}^{[k+2]} \cap \mathcal{T}^{[k+3]}].
\end{aligned}$$

Continuing in the same manner as above, we get,

$$\begin{aligned}
\mathcal{T}^{[k]} &\subseteq \mathcal{T}_{(i),u} \cup \mathcal{T}_{(i+1),u} \cup \cdots \cup \mathcal{T}_{(i+\rho-1),u} \cup [\mathcal{T}^{[k]} \cap \mathcal{T}^{[k+1]} \cap \cdots \cap \mathcal{T}^{[k+\rho]}] \\
&= \mathcal{T}_{(i),u} \cup \mathcal{T}_{(i+1),u} \cup \cdots \cup \mathcal{T}_{(i+\rho-1),u} \qquad (2.33)
\end{aligned}$$

The last line is because $\mathcal{T}^{[k]} \cap \mathcal{T}^{[k+\rho]} = \emptyset$ by Model 2.2.3. ⊠

2.B Appendix B:

Proof of Lemma 2.6.14 (bound on $\zeta^+_{j,\text{new},k}$) and of Lemma 2.7.8

Proof of Lemma 2.6.14. This proof's approach is similar to that of [11, Lemma 6.1]. The details have some differences because our main result now uses different assumptions.

This lemma uses Model 2.5.1. As shown in Lemma 2.5.2, Model 2.2.3 is a special case of this general model.

Recall that $\zeta^+_{j,\text{new},k} := \dfrac{b_{\boldsymbol{\mathcal{H}},k}}{b_{\boldsymbol{A}} - b_{\boldsymbol{A},\perp} - b_{\boldsymbol{\mathcal{H}},k}}$ with the terms on the RHS defined in Lemmas 2.6.21, 2.6.22, 2.6.23.

Recall that $\epsilon = 0.01 r_{\text{new}} \zeta \hat{\lambda}^-_{\text{train}}$. Divide the numerator and denominator by $\hat{\lambda}^-_{\text{train}}$. Define

$$B_k := \begin{cases} \left[\rho^2 h^+ (\phi^+)^2 (\kappa^+_s)^2 (\zeta^+_{j,\text{new},k-1}) + 2\kappa^+_s \phi^+\right] \left(\frac{\lambda^+_{\text{new}}}{\hat{\lambda}^-_{\text{train}}}\right) & k = 1 \\ \left[\rho^2 h^+ (\phi^+)^2 \zeta^+_{j,\text{new},k-1} + 2\sqrt{\rho^2 h^+}\phi^+\right] \left(\frac{\lambda^+_{\text{new}}}{\hat{\lambda}^-_{\text{train}}}\right) & k \geq 2 \end{cases}$$

$$C_k := \left[\rho^2 h^+ (\phi^+)^2 (\zeta^+_{j,*}) r + 2\sqrt{\rho^2 h^+}\phi^+ (\zeta^+_{j,*}) r + 2(\zeta^+_{j,*}) r\right] \left(\frac{\lambda^+}{\hat{\lambda}^-_{\text{train}}}\right) + 0.05$$

$$D_k := 1 - (\zeta^+_{j,*})^2 - (\zeta^+_{j,*})^2 \left(\frac{\lambda^+}{\hat{\lambda}^-_{\text{train}}}\right) - \zeta^+_{j,\text{new},k-1} B_k - r_{\text{new}} \zeta (C_k + .02)$$

Then,

$$\zeta^+_{j,\text{new},k} \leq \zeta^+_{j,\text{new},k-1} \frac{B_k}{D_k} + r_{\text{new}} \zeta \frac{C_k}{D_k}.$$

Recall that $\kappa^+_s = 0.0215$ and $\phi^+ = 1.2$. It is not difficult to see that $\zeta^+_{j,\text{new},k}$ is an increasing function of $\rho^2 h^+$, r, ζ, $\zeta \frac{\lambda^+}{\hat{\lambda}^-_{\text{train}}}$, and $\frac{\lambda^+_{\text{new}}}{\hat{\lambda}^-_{\text{train}}}$ and $\zeta^+_{j,\text{new},k-1}$. Consider $k = 1$. Using $\zeta^+_{j,\text{new},0} = 1$ and the upper bounds assumed in Theorem 2.2.7 on the above quantities, we get that $\zeta^+_{j,\text{new},1} \leq 0.18$.

Thus, $\zeta^+_{j,\text{new},1} \leq \zeta^+_{j,\text{new},0} = 1$. Using this and the fact that $\zeta^+_{j,\text{new},k}$ is an increasing function of $\zeta^+_{j,\text{new},k-1}$, we can show by induction that $\zeta^+_{j,\text{new},k} \leq \zeta^+_{j,\text{new},k-1}$. Thus, $\zeta^+_{j,\text{new},k} \leq \zeta^+_{j,\text{new},1} \leq 0.18$ for all $k = 1, 2 \ldots K$.

Using $\zeta^+_{j,\text{new},k} \leq 0.18$ and the bounds assumed in Theorem 2.2.7 on the other quantities we get that

$$\zeta^+_{j,\text{new},k} \leq 0.83 \zeta^+_{j,\text{new},k-1} + 0.14 r_{\text{new}} \zeta$$

Using this, we get

$$\begin{aligned}\zeta^+_{j,\text{new},k} \leq 0.83\zeta^+_{j,\text{new},k-1} + 0.14 r_{\text{new}}\zeta &\leq \zeta^+_{j,\text{new},0}(0.83)^k + \sum_{i=0}^{k-1}(0.83)^i(0.14)r_{\text{new}}\zeta \\ &\leq \zeta^+_{j,\text{new},0}(0.83)^k + \sum_{i=0}^{\infty}(0.83)^i(0.14)r_{\text{new}}\zeta \\ &\leq 0.83^k + 0.84 r_{\text{new}}\zeta.\end{aligned}$$

⊠

Proof of Lemma 2.7.8. Recall that $\boldsymbol{D}_{j,\text{new}} = (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{P}_{(j),\text{new}}$. Then $\|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{D}_{j,\text{new}}\|_2 = \|\boldsymbol{I}_{\mathcal{T}}{}'(\boldsymbol{I}-\hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{P}_{(j),\text{new}}\|_2 \leq \|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{P}_{(j),\text{new}}\|_2 + \|\hat{\boldsymbol{P}}_{(j),*}{}'\boldsymbol{P}_{(j),\text{new}}\|_2 \leq \kappa_s(\boldsymbol{P}_{(j),\text{new}}) + \|\hat{\boldsymbol{P}}_{(j),*}{}'(\boldsymbol{I}-\boldsymbol{P}_{(j),*}\boldsymbol{P}_{(j),*}{}')\boldsymbol{P}_{(j),\text{new}}\|_2 \leq \kappa_s(\boldsymbol{P}_{(j),\text{new}}) + \zeta_{j,*}$. The event $X_{\hat{u}_j+k-1} \in \Gamma^{\hat{u}_j}_{j,k-1}$ implies that $\zeta_{j,*} \leq \zeta^+_{j,*} \leq 0.0015$. Thus, the lemma follows. ⊠

2.C Appendix C: Proof of the Compressed Sensing (CS) Lemma (Lemma 2.6.15)

This proof's approach is similar to that of [11, Lemma 6.4]. The details have some differences because our main result now uses different assumptions. The proof uses the denseness assumption and subspace error bounds $\zeta_{j,*} \leq \zeta_{j,*}^{+}$ and $\zeta_{j,\text{new},k-1} \leq \zeta_{j,\text{new},k-1}^{+}$, that hold when $X_{\hat{u}_j+k-1} \in \Gamma_{j,k-1}^{\hat{u}_j}$ for $\hat{u}_j = u_j$ or $\hat{u}_j = u_j + 1$, to obtain bounds on the restricted isometry constant (RIC) of the sparse recovery matrix $\boldsymbol{\Phi}_t$ and the sparse recovery error $\|\boldsymbol{b}_t\|_2$. Applying the noisy compressed sensing (CS) result from [19] and the assumed bounds on ζ and γ, the lemma follows.

Lemma 2.C.1. *[11, Lemma 2.10] Suppose that $\boldsymbol{P}$, $\hat{\boldsymbol{P}}$ and $\boldsymbol{Q}$ are three basis matrices. Also, $\boldsymbol{P}$ and $\hat{\boldsymbol{P}}$ are of the same size, $\boldsymbol{Q}'\boldsymbol{P} = \boldsymbol{0}$ and $\|(\boldsymbol{I} - \hat{\boldsymbol{P}}\hat{\boldsymbol{P}}')\boldsymbol{P}\|_2 = \zeta_*$. Then,*

1. $\|(\boldsymbol{I} - \hat{\boldsymbol{P}}\hat{\boldsymbol{P}}')\boldsymbol{P}\boldsymbol{P}'\|_2 = \|(\boldsymbol{I} - \boldsymbol{P}\boldsymbol{P}')\hat{\boldsymbol{P}}\hat{\boldsymbol{P}}'\|_2 = \|(\boldsymbol{I} - \boldsymbol{P}\boldsymbol{P}')\hat{\boldsymbol{P}}\|_2 = \|(\boldsymbol{I} - \hat{\boldsymbol{P}}\hat{\boldsymbol{P}}')\boldsymbol{P}\|_2 = \zeta_*$
2. $\|\boldsymbol{P}\boldsymbol{P}' - \hat{\boldsymbol{P}}\hat{\boldsymbol{P}}'\|_2 \leq 2\|(\boldsymbol{I} - \hat{\boldsymbol{P}}\hat{\boldsymbol{P}}')\boldsymbol{P}\|_2 = 2\zeta_*$
3. $\|\hat{\boldsymbol{P}}'\boldsymbol{Q}\|_2 \leq \zeta_*$
4. $\sqrt{1-\zeta_*^2} \leq \sigma_i\left((\boldsymbol{I} - \hat{\boldsymbol{P}}\hat{\boldsymbol{P}}')\boldsymbol{Q}\right) \leq 1$

We begin by first bounding the RIC of the CS matrix $\boldsymbol{\Phi}_t$. We will use the notation $\kappa_s^2(\boldsymbol{P})$ to mean $(\kappa_s(\boldsymbol{P}))^2$.

Lemma 2.C.2 (Bounding the RIC of $\boldsymbol{\Phi}_t$ [11, Lemma 6.6])**.** *Recall that $\zeta_{j,*} := \|(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}')\boldsymbol{P}_{(j),*}\|_2$. The following hold.*

1. *Suppose that a basis matrix $\boldsymbol{P}$ can be split as $\boldsymbol{P} = [\boldsymbol{P}_1\ \boldsymbol{P}_2]$ where $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ are also basis matrices. Then $\kappa_s^2(\boldsymbol{P}) = \max_{\mathcal{T}:|\mathcal{T}|\leq s} \|\boldsymbol{I}_{\mathcal{T}}'\boldsymbol{P}\|_2^2 \leq \kappa_s^2(\boldsymbol{P}_1) + \kappa_s^2(\boldsymbol{P}_2)$.*
2. *$\kappa_s^2(\hat{\boldsymbol{P}}_{(j),*}) \leq (\kappa_{s,*})^2 + 2\zeta_*$ for all j*
3. *$\kappa_s(\hat{\boldsymbol{P}}_{(j),\text{new},k}) \leq \kappa_{s,\text{new}} + \zeta_{j,\text{new},k} + \zeta_{j,*}$ for all j and k.*
4. *For $t \in [(u_{j-1}+K)\alpha + 1, (\hat{u}_j+1)\alpha)$, $\delta_s(\boldsymbol{\Phi}_t) = \kappa_s^2(\hat{\boldsymbol{P}}_{(j),*}) \leq (\kappa_{s,*})^2 + 2\zeta_{j,*}$.*

5. *For* $k = 1, \ldots, K-1$, *for* $t \in [(\hat{u}_j+k)\alpha+1, (\hat{u}_j+k+1)\alpha]$ $\delta_s(\boldsymbol{\Phi}_t) = \kappa_s^2([\hat{\boldsymbol{P}}_{(j),*}\ \hat{\boldsymbol{P}}_{(j),\text{new},k}]) \leq \kappa_s^2(\hat{\boldsymbol{P}}_{(j),*}) + \kappa_s^2(\hat{\boldsymbol{P}}_{(j),\text{new},k}) \leq (\kappa_{s,*})^2 + 2\zeta_{j,*} + (\kappa_{s,\text{new}} + \zeta_{j,\text{new},k} + \zeta_{j,*})^2$.

Proof. 1. Recall that $\kappa_s^2(\boldsymbol{P}) = \max_{|\mathcal{T}|\leq s} \|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{P}\|_2^2$. Also, $\|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{P}\|_2^2 = \|\boldsymbol{I}_{\mathcal{T}}{}'[\boldsymbol{P}_1\ \boldsymbol{P}_2][\boldsymbol{P}_1\ \boldsymbol{P}_2]'\boldsymbol{I}_{\mathcal{T}}\|_2 = \|\boldsymbol{I}_{\mathcal{T}}{}'(\boldsymbol{P}_1\boldsymbol{P}_1' + \boldsymbol{P}_2\boldsymbol{P}_2')\boldsymbol{I}_{\mathcal{T}}\|_2 \leq \|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{P}_1\boldsymbol{P}_1'\boldsymbol{I}_{\mathcal{T}}\|_2 + \|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{P}_2\boldsymbol{P}_2'\boldsymbol{I}_{\mathcal{T}}\|_2$. Thus, the inequality follows.

2. For any set $\mathcal{T}$ with $|\mathcal{T}| \leq s$, $\|\boldsymbol{I}_{\mathcal{T}}{}'\hat{\boldsymbol{P}}_{(j),*}\|_2^2 = \|\boldsymbol{I}_{\mathcal{T}}{}'\hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}'\boldsymbol{I}_{\mathcal{T}}\|_2 = \|\boldsymbol{I}_{\mathcal{T}}{}'(\hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}' - \boldsymbol{P}_{(j),*}\boldsymbol{P}_{(j),*}{}' + \boldsymbol{P}_{(j),*}\boldsymbol{P}_{(j),*}{}')\boldsymbol{I}_{\mathcal{T}}\|_2 \leq \|\boldsymbol{I}_{\mathcal{T}}{}'(\hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}' - \boldsymbol{P}_{(j),*}\boldsymbol{P}_{(j),*}{}')\boldsymbol{I}_{\mathcal{T}}\|_2 + \|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{P}_{(j),*}\boldsymbol{P}_{(j),*}{}'\boldsymbol{I}_{\mathcal{T}}\|_2 \leq 2\zeta_{j,*} + (\kappa_{s,*})^2$. The last inequality follows using Lemma 2.C.1 with $\boldsymbol{P} = \boldsymbol{P}_{(j),*}$ and $\hat{\boldsymbol{P}} = \hat{\boldsymbol{P}}_{(j),*}$.

3. By Lemma 2.C.1 with $\boldsymbol{P} = \boldsymbol{P}_{(j),*}$, $\hat{\boldsymbol{P}} = \hat{\boldsymbol{P}}_{(j),*}$ and $\boldsymbol{Q} = \boldsymbol{P}_{(j),\text{new}}$, $\|\boldsymbol{P}_{(j),\text{new}}{}'\hat{\boldsymbol{P}}_{(j),*}\|_2 \leq \zeta_{j,*}$. By Lemma 2.C.1 with $\boldsymbol{P} = \boldsymbol{P}_{(j),\text{new}}$ and $\hat{\boldsymbol{P}} = \hat{\boldsymbol{P}}_{(j),\text{new},k}$, $\|(\boldsymbol{I} - \boldsymbol{P}_{(j),\text{new}}\boldsymbol{P}_{(j),\text{new}}{}')\hat{\boldsymbol{P}}_{(j),\text{new},k}\|_2 = \|(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),\text{new},k}\hat{\boldsymbol{P}}_{(j),\text{new},k}{}')\boldsymbol{P}_{(j),\text{new}}\|_2$.

   For any set $\mathcal{T}$ with $|\mathcal{T}| \leq s$, $\|\boldsymbol{I}_{\mathcal{T}}{}'\hat{\boldsymbol{P}}_{(j),\text{new},k}\|_2 \leq \|\boldsymbol{I}_{\mathcal{T}}{}'(\boldsymbol{I} - \boldsymbol{P}_{(j),\text{new}}\boldsymbol{P}_{(j),\text{new}}{}')\hat{\boldsymbol{P}}_{(j),\text{new},k}\|_2 + \|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{P}_{(j),\text{new}}\boldsymbol{P}_{(j),\text{new}}{}'\hat{\boldsymbol{P}}_{(j),\text{new},k}\|_2 \leq \|(\boldsymbol{I} - \boldsymbol{P}_{(j),\text{new}}\boldsymbol{P}_{(j),\text{new}}{}')\hat{\boldsymbol{P}}_{(j),\text{new},k}\|_2 + \|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{P}_{(j),\text{new}}\|_2 = \|(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),\text{new},k}\hat{\boldsymbol{P}}_{(j),\text{new},k}{}')\boldsymbol{P}_{(j),\text{new}}\|_2 + \|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{P}_{(j),\text{new}}\|_2 \leq \|\boldsymbol{D}_{(j),\text{new},k}\|_2 + \|\hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}'\boldsymbol{P}_{(j),\text{new}}\|_2 + \|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{P}_{(j),\text{new}}\|_2$. Taking max over $|\mathcal{T}| \leq s$ the claim follows.

4. This follows using Lemma 2.2.9 and the second claim of this lemma.

5. This follows using Lemma 2.2.9 and the first three claims of this lemma.

⊠

Corollary 2.C.3.

1. *Conditioned on* $\Gamma_{j-1,\text{end}}$, *for* $t \in [t_j, (\hat{u}_j + 1)\alpha]$, $\delta_s(\boldsymbol{\Phi}_t) \leq \delta_{2s}(\boldsymbol{\Phi}_t) \leq (\kappa_{2s,*})^2 + 2\zeta_{j,*}^+ < 0.1 < 0.1479$, *and* $\|[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\|_2 \leq \frac{1}{1-\delta_s(\boldsymbol{\Phi}_t)} < 1.2 := \phi^+$.

2. *For* $k = 2, \ldots, K$ *and* $\hat{u}_j = u_j$ *or* $\hat{u}_j = u_j+1$, *conditioned on* $\Gamma_{j,k-1}^{\hat{u}_j}$, *for* $t \in [(\hat{u}_j+k-1)\alpha+1, (\hat{u}_j + k)\alpha]$, $\delta_s(\boldsymbol{\Phi}_t) \leq \delta_{2s}(\boldsymbol{\Phi}_t) \leq (\kappa_{2s,*})^2 + 2\zeta_{j,*}^+ + (\kappa_{2s,\text{new}} + \zeta_{j,\text{new},k-1}^+ + \zeta_{j,*}^+)^2 < 0.1479$, *and* $\|[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\|_2 \leq \frac{1}{1-\delta_s(\boldsymbol{\Phi}_t)} < 1.2 := \phi^+$.

3. *For* $\hat{u}_j = u_j$ *or* $\hat{u}_j = u_j + 1$, *conditioned on* $\Gamma_{j,K}^{\hat{u}_j}$, *for* $t \in [(\hat{u}_j + K)\alpha + 1, t_{j+1} - 1]$, $\delta_s(\boldsymbol{\Phi}_t) \le \delta_{2s}(\boldsymbol{\Phi}_t) \le (\kappa_{2s,*})^2 + 2\zeta_{j,*}^+ < 0.1 < 0.1479$, *and* $\|[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\|_2 \le \frac{1}{1-\delta_s(\boldsymbol{\Phi}_t)} < 1.2 := \phi^+$.

Proof. This follows using Lemma 2.C.2, the definitions of $\Gamma_{j-1,\text{end}}$ and $\Gamma_{j,k}^{\hat{u}_j}$, and the bound on $\zeta_{j,\text{new},k-1}^+$ from Lemma 2.6.14. ⊠

The following are straightforward bounds that will be useful for the proof of Lemma 2.6.15.

Fact 2.C.4. *Under the assumptions of Theorem 2.2.7:*

- $\zeta_{j,*}^+ \gamma \le \frac{\sqrt{\zeta}}{\sqrt{r_0 + (J-1)c}} \le \sqrt{\zeta}$
- $\zeta_{j,\text{new},k-1}^+ \le 0.83^{k-1} + 0.84 r_{\text{new}} \zeta$ *(from Lemma 2.6.14)*
- $\zeta_{j,\text{new},k-1}^+ \gamma_{\text{new}} \le 0.83^{k-1}\gamma_{\text{new}} + 0.84 r_{\text{new}} \zeta \gamma_{\text{new}} \le 0.83^{k-1}\gamma_{\text{new}} + 0.3\sqrt{\zeta}$

Proof of Lemma 2.6.15. We will prove claim 2). The others are done in the same way.

Recall that $\Gamma_{j,k-1}^{\hat{u}_j}$ implies that $\zeta_{j,*} \le \zeta_{j,*}^+$ and $\zeta_{j,\text{new},k-1} \le \zeta_{j,\text{new},k-1}^+$.

a) For $t \in [(\hat{u}_j + k - 1)\alpha + 1, (\hat{u}_j + k)\alpha]$, $\boldsymbol{b}_t := (\boldsymbol{I} - \hat{\boldsymbol{P}}_{t-1}\hat{\boldsymbol{P}}_{t-1}')\boldsymbol{\ell}_t = \boldsymbol{D}_{j,*,k-1}\boldsymbol{a}_{t,*} + \boldsymbol{D}_{j,\text{new},k-1}\boldsymbol{a}_{t,\text{new}}$. Thus, using Fact 2.C.4

$$\begin{aligned}
\|\boldsymbol{b}_t\|_2 &\le \zeta_{j,*}\sqrt{r}\gamma + \zeta_{j,\text{new},k-1}\sqrt{r_{\text{new}}}\gamma_{\text{new}} \\
&\le \sqrt{\zeta}\sqrt{r} + (0.83^{k-1}\gamma_{\text{new}} + 0.84\sqrt{\zeta})\sqrt{r_{\text{new}}} \\
&= \sqrt{r_{\text{new}}}0.83^{k-1}\gamma_{\text{new}} + \sqrt{\zeta}(\sqrt{r} + 0.84\sqrt{r_{\text{new}}}) \le \xi.
\end{aligned}$$

b) By Corollary 2.C.3, $\delta_{2s}(\boldsymbol{\Phi}_t) < 0.15 < \sqrt{2} - 1$. Given $|\mathcal{T}_t| \le s$, $\|\boldsymbol{b}_t\|_2 \le \xi$, by the theorem in [19], the CS error satisfies

$$\|\hat{\boldsymbol{x}}_{t,\text{cs}} - \boldsymbol{x}_t\|_2 \le \frac{4\sqrt{1 + \delta_{2s}(\boldsymbol{\Phi}_t)}}{1 - (\sqrt{2}+1)\delta_{2s}(\boldsymbol{\Phi}_t)}\xi < 7\xi.$$

c) Using the above, $\|\hat{\boldsymbol{x}}_{t,\text{cs}} - \boldsymbol{x}_t\|_\infty \le 7\xi$. Since $\min_{i\in\mathcal{T}_t} |(\boldsymbol{x}_t)_i| \ge x_{\min}$ and $(\boldsymbol{x}_t)_{\mathcal{T}_t^c} = 0$, $\min_{i\in\mathcal{T}_t} |(\hat{\boldsymbol{x}}_{t,\text{cs}})_i| \ge x_{\min} - 7\xi$ and $\max_{i\in\bar{\mathcal{T}}_t} |(\hat{\boldsymbol{x}}_{t,\text{cs}})_i| \le 7\xi$. If $\omega < x_{\min} - 7\xi$, then $\hat{\mathcal{T}}_t \supseteq \mathcal{T}_t$. On the other hand, if $\omega > 7\xi$, then $\hat{\mathcal{T}}_t \subseteq \mathcal{T}_t$. Since ω satisfies $7\xi \le \omega \le x_{\min} - 7\xi$, the support of $\boldsymbol{x}_t$ is exactly recovered, i.e. $\hat{\mathcal{T}}_t = \mathcal{T}_t$.

d) Given $\hat{\mathcal{T}}_t = \mathcal{T}_t$, the least squares estimate of $\boldsymbol{x}_t$ satisfies $(\hat{\boldsymbol{x}}_t)_{\mathcal{T}_t} = [(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^\dagger \boldsymbol{y}_t = [(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^\dagger (\boldsymbol{\Phi}_t \boldsymbol{x}_t + \boldsymbol{\Phi}_t \boldsymbol{\ell}_t)$ and $(\hat{\boldsymbol{x}}_t)_{\bar{\mathcal{T}}_t} = \boldsymbol{0}$. Also, $(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}' \boldsymbol{\Phi}_t = \boldsymbol{I}_{\mathcal{T}_t}{}' \boldsymbol{\Phi}_t$ (this follows since $(\boldsymbol{\Phi}_t)_{\mathcal{T}_t} = \boldsymbol{\Phi}_t \boldsymbol{I}_{\mathcal{T}_t}$ and $\boldsymbol{\Phi}_t{}' \boldsymbol{\Phi}_t = \boldsymbol{\Phi}_t$). Using this, the error $\boldsymbol{e}_t := \hat{\boldsymbol{x}}_t - \boldsymbol{x}_t$ satisfies (2.18). Thus, using Fact 2.C.4 and the bounds on $\|\boldsymbol{a}_t\|_\infty$ and $\|\boldsymbol{a}_{t,\text{new}}\|_\infty$, for $t \in [(\hat{u}_j + k - 1)\alpha + 1, (\hat{u}_j + k)\alpha]$,

$$\|\boldsymbol{e}_t\|_2 \leq \phi^+(\zeta^+_{j,*}\sqrt{r}\gamma + \zeta^+_{j,\text{new},k-1}\sqrt{r_{\text{new}}}\gamma_{\text{new}}) \leq 1.2\left(1.06\sqrt{\zeta} + (0.83)^{k-1}\sqrt{r_{\text{new}}}\gamma_{\text{new}}\right)$$

The last inequality follows from Lemma 2.6.14.

$\boxtimes$

2.D Appendix D: Proof of Cauchy-Schwarz inequality for matrices

Lemma 2.D.1 (Cauchy-Schwarz for a sum of vectors)**.** *For vectors $\boldsymbol{x}_t$ and $\boldsymbol{y}_t$,*

$$\left(\sum_{t=1}^{\alpha} \boldsymbol{x}_t{}'\boldsymbol{y}_t\right)^2 \leq \left(\sum_t \|\boldsymbol{x}_t\|_2^2\right)\left(\sum_t \|\boldsymbol{y}_t\|_2^2\right)$$

Proof.

$$\left(\sum_{t=1}^{\alpha} \boldsymbol{x}_t{}'\boldsymbol{y}_t\right)^2 = \left([\boldsymbol{x}_1{}', \ldots, \boldsymbol{x}_\alpha{}'] \begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_\alpha \end{bmatrix}\right)^2 \leq \left\| \begin{bmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_\alpha \end{bmatrix} \right\|_2^2 \left\| \begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_\alpha \end{bmatrix} \right\|_2^2$$
$$= \left(\sum_{t=1}^{\alpha} \|\boldsymbol{x}_t\|_2^2\right)\left(\sum_{t=1}^{\alpha} \|\boldsymbol{y}_t\|_2^2\right)$$

The inequality is by Cauchy-Schwarz for a single vector. ⊠

Lemma 2.D.2 (Cauchy-Schwarz for a sum of matrices)**.** *For matrices $\boldsymbol{X}_t$ and $\boldsymbol{Y}_t$,*

$$\left\|\frac{1}{\alpha}\sum_{t=1}^{\alpha} \boldsymbol{X}_t\boldsymbol{Y}_t{}'\right\|_2^2 \leq \lambda_{\max}\left(\frac{1}{\alpha}\sum_{t=1}^{\alpha} \boldsymbol{X}_t\boldsymbol{X}_t{}'\right)\lambda_{\max}\left(\frac{1}{\alpha}\sum_{t=1}^{\alpha} \boldsymbol{Y}_t\boldsymbol{Y}_t{}'\right)$$

Proof of Lemma 2.D.2.

$$\left\|\sum_{t=1}^{\alpha} \boldsymbol{X}_t\boldsymbol{Y}_t{}'\right\|_2^2 = \max_{\substack{\|\boldsymbol{x}\|=1 \\ \|\boldsymbol{y}\|=1}} \left|\boldsymbol{x}'\left(\sum_t \boldsymbol{X}_t\boldsymbol{Y}_t{}'\right)\boldsymbol{y}\right|^2$$
$$= \max_{\substack{\|\boldsymbol{x}\|=1 \\ \|\boldsymbol{y}\|=1}} \left|\sum_{t=1}^{\alpha} (\boldsymbol{X}_t{}'\boldsymbol{x})'(\boldsymbol{Y}_t{}'\boldsymbol{y})\right|^2$$
$$\leq \max_{\substack{\|\boldsymbol{x}\|=1 \\ \|\boldsymbol{y}\|=1}} \left(\sum_{t=1}^{\alpha} \|\boldsymbol{X}_t{}'\boldsymbol{x}\|_2^2\right)\left(\sum_{t=1}^{\alpha} \|\boldsymbol{Y}_t{}'\boldsymbol{y}\|_2^2\right)$$
$$= \max_{\|\boldsymbol{x}\|=1} \boldsymbol{x}'\sum_{t=1}^{\alpha} \boldsymbol{X}_t\boldsymbol{X}_t{}'\,\boldsymbol{x} \cdot \max_{\|\boldsymbol{y}\|=1} \boldsymbol{y}'\sum_{t=1}^{\alpha} \boldsymbol{Y}_t\boldsymbol{Y}_t{}'\,\boldsymbol{y}$$
$$= \lambda_{\max}\left(\sum_{t=1}^{\alpha} \boldsymbol{X}_t\boldsymbol{X}_t{}'\right)\lambda_{\max}\left(\sum_{t=1}^{\alpha} \boldsymbol{Y}_t\boldsymbol{Y}_t{}'\right)$$

The inequality is by Lemma 2.D.1. The penultimate line is because $\|\boldsymbol{x}\|_2^2 = \boldsymbol{x}'\boldsymbol{x}$. Multiplying both sides by $\left(\frac{1}{\alpha}\right)^2$ gives the desired result. ⊠

References

[1] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of ACM*, vol. 58, no. 3, 2011.

[2] V. Chandrasekaran, S. Sanghavi, P. A. Parrilo, and A. S. Willsky, "Rank-sparsity incoherence for matrix decomposition," *SIAM Journal on Optimization*, vol. 21, 2011.

[3] D. Hsu, S. Kakade, and T. Zhang, "Robust matrix decomposition with sparse corruptions," *IEEE Trans. Info. Th.*, Nov. 2011.

[4] M. Fazel, "Matrix rank minimization with applications," *PhD thesis, Stanford University*, 2002.

[5] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Found. of Comput. Math*, no. 9, pp. 717–772, 2008.

[6] M. Brand, "Incremental singular value decomposition of uncertain data with missing values," in *Eur. Conf. on Comp. Vis. (ECCV)*, 2002.

[7] L. Balzano, B. Recht, and R. Nowak, "Online Identification and Tracking of Subspaces from Highly Incomplete Information," in *Allerton Conf. Communication, Control, and Computing*, 2010.

[8] Y. Chi, Y. C. Eldar, and R. Calderbank, "Petrels: Parallel subspace estimation and tracking by recursive least squares from partial observations," *IEEE Trans. Sig. Proc.*, December 2013.

[9] L. Balzano and S. Wright, "Local convergence of an algorithm for subspace identification from partial data," *arXiv:1306.3391 [cs.NA]*.

[10] A. Krishnamurthy and A. Singh, "Low-rank matrix and tensor completion via adaptive sampling," in *Advances in Neural Information Processing Systems 26*, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 836–844. [Online]. Available: http://papers.nips.cc/paper/4954-low-rank-matrix-and-tensor-completion-via-adaptive-sampling.pdf

[11] C. Qiu, N. Vaswani, B. Lois, and L. Hogben, "Recursive robust pca or recursive sparse recovery in large but structured noise," *IEEE Trans. Info. Th.*, Aug. 2014, shorter versions in ICASSP 2013 and ISIT 2013.

[12] J. Feng, H. Xu, and S. Yan, "Online robust pca via stochastic optimization," in *Adv. Neural Info. Proc. Sys. (NIPS)*, 2013.

[13] J. Feng, H. Xu, S. Mannor, and S. Yan, "Online pca for contaminated data," in *Advances in Neural Information Processing Systems 26*, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 764–772. [Online]. Available: http://papers.nips.cc/paper/5135-online-pca-for-contaminated-data.pdf

[14] J. He, L. Balzano, and A. Szlam, "Incremental gradient on the grassmannian for online foreground and background separation in subsampled video," in *IEEE Conf. on Comp. Vis. Pat. Rec. (CVPR)*, 2012.

[15] P. Netrapalli, P. Jain, and S. Sanghavi, "Low-rank matrix completion using alternating minimization," in *Symposium on Theory of Computing (STOC)*, 2013.

[16] B. Lois and N. Vaswani, "A correctness result for online robust pca," in *IEEE Intl. Conf. Acoustics, Speech, Sig. Proc. (ICASSP)*, 2015.

[17] H. Guo, C. Qiu, and N. Vaswani, "An online algorithm for separating sparse and low-dimensional signal sequences from their sum," *IEEE Trans. Sig. Proc.*, pp. 4284–4297, Aug. 2014.

[18] F. D. L. Torre and M. J. Black, "A framework for robust subspace learning," *International Journal of Computer Vision*, vol. 54, pp. 117–142, 2003.

[19] E. Candes, "The restricted isometry property and its implications for compressed sensing," *Compte Rendus de l'Academie des Sciences, Paris, Serie I*, pp. 589–592, 2008.

[20] J. Feng, H. Xu, and S. Yan, "Online robust pca via stochastic optimization," in *Advances in Neural Information Processing Systems 26*, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 404–412. [Online]. Available: http://papers.nips.cc/paper/5131-online-robust-pca-via-stochastic-optimization.pdf

[21] J. Zhan and N. Vaswani, "Robust pca with partial subspace knowledge," in *IEEE Intl. Symp. Info. Th. (ISIT)*, 2014.

[22] C. Davis and W. M. Kahan, "The rotation of eigenvectors by a perturbation. iii," *SIAM Journal on Numerical Analysis*, Mar. 1970.

[23] J. A. Tropp, "User-friendly tail bounds for sums of random matrices," *Foundations of Computational Mathematics*, vol. 12, no. 4, 2012.

[24] R. Horn and C. Johnson, *Matrix Analysis*. Cambridge Univ. Press, 1985.

[25] M. Muselli, "On convergence properties of pocket algorithm," *Neural Networks, IEEE Transactions on*, vol. 8, no. 3, pp. 623–629, May 1997.

[26] J. Zhan and N. Vaswani, "Performance guarantees for reprocs – correlated low-rank matrix entries case," arXiv:1405.5887 [cs.IT].

[27] B. Lois, N. Vaswani, and C. Qiu, "Performance guarantees for undersampled recursive sparse recovery in large but structured noise," in *GlobalSIP*, 2013.

CHAPTER 3. RECURSIVE SPARSE RECOVERY IN CORRELATED STRUCTURED NOISE

A paper prepared for submission to *IEEE Transactions on Information Theory*

Brian Lois, Namrata Vaswani, and Jinchun Zhan

Abstract

This work studies the problem of recursive robust principal components analysis (PCA). At each time t, suppose that a vector $\boldsymbol{m}_t = \boldsymbol{\ell}_t + \boldsymbol{x}_t$ is observed. The vectors $\boldsymbol{\ell}_t$ lie in a slowly changing and dense low-dimensional subspace. The vectors $\boldsymbol{x}_t$ are sparse and their support changes frequently enough. The goal is to recover $\boldsymbol{x}_t$ and $\boldsymbol{\ell}_t$ at each time t and to maintain an estimate of range$(\boldsymbol{\ell}_1, \dots, \boldsymbol{\ell}_t)$. This work improves on existing results by assuming a more general autoregressive model for the low-dimensional vectors $\boldsymbol{\ell}_t$. It is proven that under certain model assumptions, with high probability, the practical ReProCS algorithm exactly recovers the support of $\boldsymbol{x}_t$, and the error made in estimating $\boldsymbol{x}_t$ and $\boldsymbol{\ell}_t$ is bounded by a small value that depends on the accuracy of the initial subspace estimate. Also, all of the subspace changes are detected within a certain delay, and the error made in estimating the new subspace decays below a small value within a finite delay.

3.1 Introduction

Principal Components Analysis (PCA) is a tool that is frequently used for dimension reduction. Given a matrix of data $\boldsymbol{D}$, PCA seeks to recover a small number of directions that contain most of the information in data. This is typically accomplished by performing a singular value decomposition (SVD) of $\boldsymbol{D}$ and retaining the singular vectors corresponding to the largest singular values. A limitation of this procedure is that it is highly sensitive to outliers

in the data set. Recently there has been much work done to develop and analyze algorithms for PCA that are robust with respect to outliers. A common way to model outliers is as sparse vectors [1]. In seminal papers Candès et. al. and Chandrasekaran et. al. introduced the Principal Components Pursuit (PCP) convex program and proved its robustness to sparse outliers [2], [3]. Principal Components Pursuit poses the robust PCA problem as identifying a low rank matrix and a sparse matrix from their sum. The program is to minimize a weighted sum of the nuclear norm of the low rank matrix and the vector ℓ_1 norm of the sparse matrix subject to their sum being equal to the observed data matrix. The results in [4] improve upon those in [3]. Other methods such as [5] model the entire column vector as being either correct or an outlier. Some other works on the performance guarantees for batch robust PCA include [6], [7], and [8]. All of these methods require waiting until all of the data has been acquired before performing the optimization.

In this work we consider an online or recursive version of the robust PCA problem where we seek to separate vectors into low dimensional and sparse components as they arrive, using the previous estimates, rather than re-solving the entire problem at each time t. An application where this type of problem is useful is in video analysis [9]. Imagine a video sequence that has a distinct background and foreground. An example might be a surveillance camera where a person walks across the scene. If the background does not change very much, and the foreground is sparse (both practical assumptions), then separating the background and foreground can be viewed as a robust PCA problem. Sparse plus low rank decomposition can also be used to detect anomalies in network traffic patterns [10]. In all such an applications an online solution is desirable.

3.1.1 Paper Organization

This paper is organized in the following way. The signal model, assumptions, and main result are given in Section 3.2. Here we explain special cases of support change of $\boldsymbol{x}_t$ that will satisfy our assumptions. Section 3.3 contains our most general support change model. Proofs of the support change results can be found in Appendix 3.B. A description of the Algorithm studied is in Section 3.4.

The main theorem (Theorem 3.2.15) is proved in Section 3.5. Proofs of the main lemmas used to prove the theorem are given next in Sections 3.6 and 3.7. Some preliminary and simple results used in the proofs are deferred to the Appendix. Appendix 3.A contains lemmas for exchanging the order of a double sum (Lemma 3.A.1), Cauchy-Schwarz for matrices (Lemma 3.A.3), and the matrix Azuma inequality from [11] and associated corollaries (Lemmas 3.A.8 - 3.A.10).

In Section 3.8 we introduce a more general subspace change model and a corresponding algorithm. A result analogous to Theorem 3.2.15 is proven in Section 3.9. Finally, in Section 3.10 we provide some simple simulation results that demonstrate the theoretical results.

3.1.2 Problem Definition

At time t we observe a vector $\boldsymbol{m}_t \in \mathbb{R}^n$ that is the sum of a vector from a slowly changing low-dimensional subspace $\boldsymbol{\ell}_t$ and a sparse vector $\boldsymbol{x}_t$. So

$$\boldsymbol{m}_t = \boldsymbol{\ell}_t + \boldsymbol{x}_t \qquad \text{for } t = 1, 2, \ldots, t_{\max},$$

with the possibility that $t_{\max} = \infty$. We model the low-dimensional $\boldsymbol{\ell}_t$'s as $\boldsymbol{\ell}_t = \boldsymbol{P}_t \boldsymbol{a}_t$ for a basis matrix $\boldsymbol{P}_t$ that is allowed to change slowly over time. Given an estimate of the initial subspace $\hat{\boldsymbol{P}}_{(0)}$, the goal is to obtain estimates $\hat{\boldsymbol{x}}_t$ and $\hat{\boldsymbol{\ell}}_t$ at each time t and to periodically update the estimate of $\boldsymbol{P}_t$.

3.1.3 Contribution

The ReProCS algorithm for online sparse + low-rank matrix recovery (online robust PCA) was first introduced and analyzed in [12]. That paper contained a result that assumed certain properties of the algorithm estimates, and as such was only a partial correctness result. In Chapter 2, by building on proof techniques introduced in [12], a full correctness result was proven for the same ReProCS algorithm. Experimental evaluation of ReProCS is done in [14]. Here it is shown that with practical heuristics used to set its parameters (some of which are different from the algorithm parameters used in our correctness result), ReProCS has signifi-

cantly improved recovery performance compared to other recursive ([15, 16, 10]) and even batch methods ([2, 9, 16]) for many simulated and real video datasets.

A limitation of the result of Chapter 2 was that it assumed independence of the $\boldsymbol{\ell}_t$'s over time. For the video application, this means that the background images[1] are independent over time, and in most cases this is not a valid assumption. In this work, we are able to remove this assumption and replace it by a more realistic first order autoregressive (AR) model assumption on the $\boldsymbol{\ell}_t$'s. The ReProCS algorithm itself does not need knowledge of the AR model or its parameters. Under this model and only one extra assumption compared to Chapter 2, we can obtain a correctness result. The extra assumption needed is a bound on the ratio of the squared maximum value to the variance of any entry of $\boldsymbol{a}_t$ (which will later be defined as the coefficients of $\boldsymbol{\ell}_t$ with respect to an orthonormal basis). We show that as long as algorithm parameters are set appropriately, a good enough estimate of the initial subspace is available, slow subspace change holds, the subspaces are dense enough, and there is a certain amount of support change at least every so often, then the support can be exactly recovered with high probability, the sparse and low-rank matrix columns can be recovered with bounded and small error, and the subspace recovery error decays to a small value within a finite delay of a subspace change. Use of the AR model requires new proof techniques beyond what were used in Chapter 2. We need to use the matrix Azuma inequality from [11] instead of the matrix Hoeffding inequality from the same paper. Before applying the matrix Azuma inequality, we must also perform algebraic manipulation of sums so that the previous term on which we are conditioning is small. This is done in Lemma 3.6.3 which is proved using Lemma 3.A.6. Lemma 3.6.3 is used in the proof of Lemma 3.5.22 which is also significantly different.

A second contribution of this work is that we assume a subspace change model that allows for both addition of new directions and removal of existing directions from the subspace, and we also study a partly practical modification of the ReProCS with cluster-PCA (ReProCS-cPCA) algorithm introduced in [12]. ReProCS-cPCA improves upon ReProCS in that it also includes a subspace re-estimation step that allows removal of deleted directions from the subspace estimate. We obtain a correctness result for this algorithm under all the earlier assumptions

[1] technically the background image minus a mean background image

and a clustering assumption on the eigenvalues of the covariance matrix of $\boldsymbol{\ell}_t$ after the subspace change has stabilized. A key advantage of this result is that it significantly relaxes the denseness requirements and consequently needs a much looser upper bound on the rank-sparsity product compared with our result for ReProCS. The ReProCS result needs a bound that is tighter than what PCP needs (PCP is a batch method while ReProCS is online) but the bound needed by ReProCS-cPCA is comparable to that of PCP. In fact, ReProCS-cPCA does not need a bound on the rank of $\boldsymbol{L}$ as long as the delay between subspace change times increases in proportion to $\log J$ where J is the total number of subspace change times in the entire sequence and J is known. The requirement that the length of time between subspace times increases with J is a consequence of the probabilistic signal model, and not the algorithm. Another way to interpret the result is that the probability of incorrect recovery increases only linearly with J. However this result has a significant limitation. Unlike the practical ReProCS algorithm, the ReProCS-cPCA algorithm is not fully automatic. It needs information about the eigenvalue clustering which at this point we cannot set automatically. (See the discussion in Section 3.8.3.)

To the best of our knowledge, this is among the first few works that provides a correctness result for an online (recursive) algorithm for sparse plus low-rank matrix recovery or equivalently for online robust PCA. As an easy corollary, we also have a result for online matrix completion. In this case, the support of $\boldsymbol{x}_t$ is the set of missing entries, and this is known. Online algorithms are needed for real-time applications; even for offline applications, they are faster and need less storage compared to batch techniques. Moreover, online approaches can provide a natural way to exploit temporal dependencies in the dataset. In our case, we show that ReProCS uses slow subspace change to allow for significantly more correlated support sets of the sparse vectors than do the various results for PCP [2, 3, 17]. Partial results have been provided for online sparse plus low-rank matrix recovery in our earlier work [12], and also in later work by Feng et. al. [18]; however, all require an assumption on intermediate algorithm estimates. We discuss these and [19, 20] in Sec 3.2.6. There is some more recent work on online robust PCA algorithms and their experimental evaluation, e.g. [15], [21].

The new proof techniques developed in this and earlier works [12, 13] are needed because in our case, the error $\boldsymbol{e}_t = \boldsymbol{\ell}_t - \hat{\boldsymbol{\ell}}_t$ is correlated with the true data $\boldsymbol{\ell}_t$. This is an artifact of how

we obtain the estimates $\boldsymbol{\ell}_t$. Standard results for PCA such as those in [22] cannot be used, because they assume that the noise is independent of or uncorrelated with the noise-free data.

3.1.4 Notation

We use lowercase bold letters for vectors and capital bold letters for matrices. We use $\boldsymbol{x}'$ for the transpose of a vector $\boldsymbol{x}$ and similarly $\boldsymbol{A}'$ for the transpose of a matrix $\boldsymbol{A}$. (Everything is real, so this is also the Hermitian adjoint). The 2-norm of a vector and the induced 2-norm of a matrix are denoted by $\|\cdot\|_2$. We refer to a matrix with orthonormal columns as a *basis matrix*. Notice that for a basis matrix $\boldsymbol{P}$, $\boldsymbol{P}'\boldsymbol{P} = \boldsymbol{I}$. For a set $\mathcal{T}$ of integers, $|\mathcal{T}|$ denotes its cardinality. For a vector $\boldsymbol{x}$, $\boldsymbol{x}_\mathcal{T}$ is a smaller vector containing the entries of $\boldsymbol{x}$ indexed by $\mathcal{T}$. Define $\boldsymbol{I}_\mathcal{T}$ to be an $n \times |\mathcal{T}|$ matrix of those columns of the identity matrix indexed by $\mathcal{T}$. Then let $\boldsymbol{A}_\mathcal{T} := \boldsymbol{A}\boldsymbol{I}_\mathcal{T}$. For matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ where the columns of $\boldsymbol{Q}$ are a subset of the columns of $\boldsymbol{P}$, we will use the notation $\boldsymbol{P}\setminus\boldsymbol{Q}$ to mean the matrix of columns in $\boldsymbol{P}$ and not in $\boldsymbol{Q}$. Using our column subscripting notation, for an $n \times r$ matrix $\boldsymbol{P}$, if $\boldsymbol{Q} = \boldsymbol{P}_\mathcal{T}$, then $\boldsymbol{P}\setminus\boldsymbol{Q} := \boldsymbol{P}_{[1,\ldots,r]\setminus\mathcal{T}} = \boldsymbol{P}_{\overline{\mathcal{T}}}$.

For integers a and b, we write $a \mod b$ for the remainder when a is divided by b. We use the interval notation $[a, b]$ to mean all of the integers between a and b, inclusive, and similarly for (a, b) etc. For a matrix $\boldsymbol{A}$, the restricted isometry constant (RIC) $\delta_s(\boldsymbol{A})$ is the smallest real number δ_s such that

$$(1 - \delta_s)\|\boldsymbol{x}\|_2^2 \leq \|\boldsymbol{A}\boldsymbol{x}\|_2^2 \leq (1 + \delta_s)\|\boldsymbol{x}\|_2^2$$

for all s-sparse vectors $\boldsymbol{x}$ [23]. A vector $\boldsymbol{x}$ is s-sparse if it has s or fewer non-zero entries. For Hermitian matrices $\boldsymbol{A}$ and $\boldsymbol{B}$, the notation $\boldsymbol{A} \preceq \boldsymbol{B}$ means that $\boldsymbol{B} - \boldsymbol{A}$ is positive semi-definite. For an Hermitian matrix $\boldsymbol{H}$, $\boldsymbol{H} \overset{\text{EVD}}{=} \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}'$ denotes its eigenvalue decomposition. Similarly for any matrix $\boldsymbol{A}$, $\boldsymbol{A} \overset{\text{SVD}}{=} \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}'$ denotes its singular value decomposition.

For basis matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$, define

$$\text{dif}(\boldsymbol{P}, \boldsymbol{Q}) := \|(\boldsymbol{I} - \boldsymbol{P}\boldsymbol{P}')\boldsymbol{Q}\|_2.$$

It is not difficult to show that this function is symmetric when $\boldsymbol{P}$ and $\boldsymbol{Q}$ are the same size. (See for example [12]). We will use the function dif as a measure of the error made when estimating

range($\boldsymbol{Q}$) by range($\boldsymbol{P}$). We use the hat to denote estimation of the column space of a matrix. So range($\hat{\boldsymbol{P}}$) is an estimate of range($\boldsymbol{P}$), but $\hat{\boldsymbol{P}}$ is not necessarily an entry-wise estimate of $\boldsymbol{P}$.

We will use $\mathbb{P}$ for probability and $\mathbb{E}$ for expectation. For a sequence of random variables $\boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_t$, the notation

$$\mathbb{E}_{t-1}[\boldsymbol{Z}_t] := \mathbb{E}[\boldsymbol{Z}_t | \boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_{t-1}]$$

and

$$\mathbb{E}_{t-1}[\boldsymbol{Z}_t | X] := \mathbb{E}[\boldsymbol{Z}_t | X, \boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_{t-1}]$$

For an event (set) Γ, we use $\overline{\Gamma}$ for the complement of Γ.

3.2 Model Assumptions, Main Result, and Discussion

3.2.1 Model on $\boldsymbol{\ell}_t$

Model 3.2.1.

1. *Subspace Change Model for $\boldsymbol{\ell}_t$*

   Let t_j for $j = 1, \ldots, J$ be the times at which the subspace changes. For the sake of notation, let $t_0 = 0$ and $t_{J+1} := t_{\max}$. We assume $\boldsymbol{\ell}_t = \boldsymbol{P}_t \boldsymbol{a}_t$ for all $t = 1, \ldots, t_{\max}$, and

$$\boldsymbol{P}_t = \begin{cases} [\boldsymbol{P}_{t-1} \ \boldsymbol{P}_{t,\text{new}}] & \text{if } t = t_1 \text{ or } t_2 \text{ or } \ldots t_J \\ \boldsymbol{P}_{t-1} & \text{otherwise} \end{cases} \tag{3.1}$$

   where $\boldsymbol{P}_t$ is a basis matrix for all t.

   Let $r_j = \text{rank}(\boldsymbol{P}_{t_j})$ and $c_{j,\text{new}} = \text{rank}(\boldsymbol{P}_{t_j,\text{new}})$.

2. *Assumptions and notation for $\boldsymbol{a}_t$. We assume the following model on $\boldsymbol{a}_t$:*

$$\boldsymbol{a}_t = b\boldsymbol{a}_{t-1} + \boldsymbol{\nu}_t$$

   for a scalar $b < 1$. Set $\boldsymbol{a}_0 = \boldsymbol{0}$. Also assume $\mathbb{E}[\boldsymbol{\nu}_t] = \boldsymbol{0}$, the $\boldsymbol{\nu}_t$ are mutually independent over t, bounded, and the matrix $\boldsymbol{\Lambda}_{\nu,t} := \text{Cov}(\boldsymbol{\nu}_t)$ is diagonal.

   At $t = t_j$ the length of the vector $\boldsymbol{a}_t$ increases from r_{j-1} to r_j. From (3.1) it is clear that for $t < t_j$, $\boldsymbol{P}_{t_j,\text{new}}{}'\boldsymbol{\ell}_t = \boldsymbol{0}$. Therefore, for the autoregressive model, the last $(r_j - r_{j-1})$ entries of $\boldsymbol{a}_t$ begin at 0.

Define

(a) $\gamma := \dfrac{\sup_t \|\boldsymbol{\nu}_t\|_\infty}{(1-b)}$ *(Since* $\boldsymbol{\nu}_t$ *is bounded,* $\gamma < \infty$*.)*

(b) $\lambda^- := \dfrac{\inf_t \lambda_{\min}(\boldsymbol{\Lambda}_{\nu,t})}{1-b^2}$ *and* $\lambda^+ := \dfrac{\sup_t \lambda_{\max}(\boldsymbol{\Lambda}_{\nu,t})}{1-b^2}$

and assume that $0 < \lambda^- \leq \lambda^+ < \infty$.

By induction it is easy to see that

$$\|\boldsymbol{a}_t\|_\infty = \|b\boldsymbol{a}_{t-1} + \boldsymbol{\nu}_t\|_\infty \leq b\|\boldsymbol{a}_{t-1}\|_\infty + \|\boldsymbol{\nu}_t\|_\infty \leq \; b\gamma + (1-b)\gamma \leq \gamma.$$

Definition 3.2.2. *Define*

$$\boldsymbol{P}_{(j)} := \boldsymbol{P}_{t_j} \text{ for } j = 1, \ldots, J$$

$$\boldsymbol{P}_{(j),*} := \boldsymbol{P}_{(j-1)}$$

$$\boldsymbol{P}_{(j),\text{new}} := \boldsymbol{P}_{t_j,\text{new}}$$

Because $\boldsymbol{P}_t$ *is a basis matrix,* $\boldsymbol{P}_{(j),*} \perp \boldsymbol{P}_{(j),\text{new}}$. *Define* $r_J = \max_j \operatorname{rank} \boldsymbol{P}_{(j)}$. *and* $c_{\text{new}} := \max_j \operatorname{rank}(\boldsymbol{P}_{(j),\text{new}})$. *Observe that*

$$r_J = \operatorname{rank}([\boldsymbol{\ell}_1, \ldots, \boldsymbol{\ell}_{t_{\max}}]) \quad \text{and} \quad r_J \leq r_0 + Jc_{\text{new}} := r.$$

For $t \in [t_j, t_{j+1})$, $\boldsymbol{\ell}_t$ *can be written as* $\boldsymbol{\ell}_t = [\boldsymbol{P}_{(j),*} \; \boldsymbol{P}_{(j),\text{new}}] \begin{bmatrix} \boldsymbol{a}_{t,*} \\ \boldsymbol{a}_{t,\text{new}} \end{bmatrix}$, *where*

$$\boldsymbol{a}_{t,*} := \boldsymbol{P}_{(j),*}{}'\boldsymbol{\ell}_t \quad \text{and} \quad \boldsymbol{a}_{t,\text{new}} := \boldsymbol{P}_{(j),\text{new}}{}'\boldsymbol{\ell}_t \tag{3.2}$$

Definition 3.2.3. *As in* (3.2), *for* $t \in [t_j, t_{j+1})$, *define*

$$\boldsymbol{\nu}_{t,*} := (\boldsymbol{\nu}_t)_{[1, r_j - c_{j,\text{new}}]} \quad \text{and} \quad \boldsymbol{\nu}_{t,\text{new}} := (\boldsymbol{\nu}_t)_{[r_j - c_{j,\text{new}} + 1, r_j]}$$

and define $\boldsymbol{\Lambda}_{\nu,t} := \operatorname{Cov}(\boldsymbol{\nu}_t)$. *Also define* $\boldsymbol{\Lambda}_{a,t} := \operatorname{Cov}(\boldsymbol{a}_t)$.

Observe that

$$\boldsymbol{\Lambda}_{a,t} = b^2\boldsymbol{\Lambda}_{a,t-1} + \boldsymbol{\Lambda}_{\nu,t} \tag{3.3}$$

From the above equation, it is clear that $\boldsymbol{\Lambda}_{a,t}$ is also diagonal and

$$(1-b^{2t})\lambda^- \leq \lambda_{\min}(\boldsymbol{\Lambda}_{a,t}) \leq \lambda_{\max}(\boldsymbol{\Lambda}_{a,t}) \leq (1-b^{2t})\lambda^+.$$

Definition 3.2.4. *Define*

$$f := \frac{\lambda^+}{\lambda^-}$$

Notice that f is a bound on the condition number of $\mathbf{\Lambda}_{a,t}$ at any time t.

Definition 3.2.5. *Define* $\mathbf{\Lambda}_{a,t,*} := \mathrm{Cov}(\boldsymbol{a}_{t,*})$ *and* $\mathbf{\Lambda}_{a,t,\mathrm{new}} := \mathrm{Cov}(\boldsymbol{a}_{t,\mathrm{new}})$. *Similarly define* $\mathbf{\Lambda}_{\nu,t,*} := \mathrm{Cov}(\boldsymbol{\nu}_{t,*})$ *and* $\mathbf{\Lambda}_{\nu,t,\mathrm{new}} := \mathrm{Cov}(\boldsymbol{\nu}_{t,\mathrm{new}})$. *Then*

$$\mathbf{\Lambda}_{a,t} = \begin{bmatrix} \mathbf{\Lambda}_{a,t,*} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_{a,t,\mathrm{new}} \end{bmatrix} \text{ and } \mathbf{\Lambda}_{\nu,t} = \begin{bmatrix} \mathbf{\Lambda}_{\nu,t,*} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_{\nu,t,\mathrm{new}} \end{bmatrix}.$$

Definition 3.2.6. *For an integer d, define*

1. $\gamma_{\mathrm{new}} := \dfrac{\max_j \max_{t\in[t_j,t_j+d]} \|\boldsymbol{\nu}_{t,\mathrm{new}}\|_\infty}{(1-b)}$
2. $\lambda^-_{\mathrm{new}} := \min_j \min_{t\in[t_j,t_j+d]} \dfrac{\lambda_{\min}(\mathbf{\Lambda}_{\nu,t,\mathrm{new}})}{1-b^2}$ *and* $\lambda^+_{\mathrm{new}} := \max_j \max_{t\in[t_j,t_j+d]} \dfrac{\lambda_{\max}(\mathbf{\Lambda}_{\nu,t,\mathrm{new}})}{1-b^2}$

In the theorem we assume upper bounds on γ_{new} and λ^+_{new}. This ensures that the projection of $\boldsymbol{\ell}_t$ along the new directions is "small" for some time (d frames) after a subspace change.

Definition 3.2.7. *Define*

$$\eta := \frac{\gamma^2}{\lambda^+} \text{ and } \eta_{\mathrm{new}} := \frac{{\gamma_{\mathrm{new}}}^2}{\lambda^+_{\mathrm{new}}}$$

Observe that if $\boldsymbol{a}_t$ is a scalar random variable, then η is the ratio of the maximum absolute value (of $\boldsymbol{a}_t$) squared to the variance. For a continuous random variable uniformly distributed on the interval $[-\gamma, \gamma]$ one can easily compute $\eta = 3$. Moreover, if the i-th entry of $\boldsymbol{a}_t$ is a continuous random variables that is uniformly distributed in $[-\gamma_i, \gamma_i]$ and if this is true for all times t, then $\gamma = \max_i \gamma_i$ and we still get $\eta = \eta_{\mathrm{new}} = 3$.

Subspace Change Model with Deletions. The above simple model only assumes new directions get added to the subspace but nothing gets removed. We work with this model for notational simplicity. Our results in this section will also hold if it is replaced by the following more general model.

Model 3.2.8. *Signal Model 3.2.1 holds with* $\boldsymbol{P}_t = [(\boldsymbol{P}_{t-1}\boldsymbol{R}_t) \setminus \boldsymbol{P}_{t,\mathrm{old}} \quad \boldsymbol{P}_{t,\mathrm{new}}]$ *at $t = t_j$'s. Here $\boldsymbol{R}_t$ is an arbitrary rotation matrix. (By using a rotation matrix, we allow the removal of any direction in* range$(\boldsymbol{P}_{t-1})$*.)*

In Section 3.8, we study a more general ReProCS algorithm called ReProCS-cPCA that includes a deletion (by subspace re-estimation) step. Our result for that algorithm is based on Signal Model 3.2.8. With one extra assumption, we are able to prove a stronger conclusion for ReProCS-cPCA.

3.2.2 Denseness coefficient

Below we give the definition of the denseness coefficient κ_s.

Definition 3.2.9. *For a basis matrix $\boldsymbol{P}$, define* $\kappa_s(\boldsymbol{P}) := \max_{|\mathcal{T}| \le s} \|\boldsymbol{I}_{\mathcal{T}}{}' \boldsymbol{P}\|_2$.

As described in [12], κ_s is a measurement of the denseness of the vectors in the subspace range($\boldsymbol{P}$). Notice that small κ_s means that the columns of $\boldsymbol{P}$ are dense vectors. The reason for quantifying denseness using κ_s is the following lemma from [12].

Lemma 3.2.10. *For a basis matrix $\boldsymbol{P}$, $\delta_s(\boldsymbol{I} - \boldsymbol{P}\boldsymbol{P}') = (\kappa_s(\boldsymbol{P}))^2$.*

3.2.3 Model on $\boldsymbol{x}_t$

Let $\mathcal{T}_t := \{i : (\boldsymbol{x}_t)_i \neq 0\}$ be the support set of $\boldsymbol{x}_t$ and let

$$s := \max_t |\mathcal{T}_t|$$

be the size of the largest support, and let

$$x_{\min} := \inf_t \min_{i \in \mathcal{T}_t} |(\boldsymbol{x}_t)_i|$$

denote the size of the smallest non-zero entry of any $\boldsymbol{x}_t$.

In order to prove our result, we require that the supports of $\boldsymbol{x}_t$ be sufficiently different. Below we give two possible models that will imply the most general conditions required (given in Section 3.3).

Model 3.2.11. *Let $\varrho \ge 1$ be a scalar. Suppose that the support of $\boldsymbol{x}_t$ is of fixed size less than or equal to s, consists of consecutive indices, and moves down the vector at least every β time instants. Moreover, each time it moves, it moves by at least $\frac{s}{\varrho}$ indices and at most $\varrho_2 s$ indices. Stated differently, the support of $\boldsymbol{x}_t$ remains the same for no more than β time instants, and when it moves, it moves by no fewer than $\frac{s}{\varrho}$ indices and no more than $\varrho_2 s$ indices.*

Model 3.2.12. *Suppose that the support of $\boldsymbol{x}_t$ consists of s or fewer consecutive indices and moves down the vector by between 1 and m indices at every time t.*

Figure 3.5 on page 98 illustrates the above support change models.

Remark 3.2.13. *For both of the above models, when the support reaches the bottom of the vector, we assume that it starts over at* 1*. This models a moving 1D object of length s or less that enters the scene and eventually walks out, and then another object of length s or less may come in. The requirement of consecutive indices and downward (as opposed to upward) motion are done for simplicity and ease of understanding. Our results still hold under permutations (relabeling) of the indices. We could also make a small modification and assume that the object is reflected back up (down) when it reaches the bottom (top). See Remark 3.3.6.*

Remark 3.2.14. *Nothing in our most general support change model or our algorithm requires only one object in the support of $\boldsymbol{x}_t$. The models above are simple examples that capture the intuition of our general model.*

3.2.4 Main Result

Theorem 3.2.15. *Consider Algorithm 3. Pick a ζ that satisfies*

$$\zeta \leq \min\left\{\frac{10^{-4}}{r^2}, \frac{1.5\times 10^{-4}}{r^2 f}, \frac{1}{r^3\gamma^2}, \frac{0.01\lambda^-}{b^2 r^3 \gamma^2}\right\}.$$

Suppose

1. $\|(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(0)}\hat{\boldsymbol{P}}_{(0)}{}')\boldsymbol{P}_{(0)}\|_2 \leq r_0\zeta$*;*
2. *The algorithm parameters are set as:*
   - $\text{thresh} = \frac{\lambda^-}{2}$*;*
   - $K = \left\lceil \frac{\log(0.16 c_{\text{new}}\zeta)}{\log(0.4)} \right\rceil$*;*
   - $\xi = \sqrt{c_{\text{new}}}\gamma_{\text{new}} + (\sqrt{r} + \sqrt{c_{\text{new}}})\sqrt{\zeta}$*;*
   - $\omega = 7\xi$*;*

- $\alpha = C(\log(35KJ) + 11\log(n))$ *for a constant* $C \geq C_{\text{add}}$ *with*

$$C_{\text{add}} := 20^2 \cdot 8 \cdot 48^2 \frac{(\phi^+ \xi)^4}{(c_{\text{new}} \zeta \lambda^-)^2}; \tag{3.4}$$

3. *Signal Model 3.2.1 on* ℓ_t *holds with* $b \leq 0.1$ *and*

   (a) $t_{j+1} - t_j > d \geq (K+2)\alpha$ *for all* j *i.e. the delay between change times is "large";*

   (b) $\boldsymbol{a}_{t,\text{new}}$ *is "small"*

      i. $\sqrt{c_{\text{new}}}\gamma_{\text{new}} + (\sqrt{r} + \sqrt{c_{\text{new}}})\sqrt{\zeta} \leq \frac{x_{\min}}{14}$;

      ii. $\lambda^+_{\text{new}} \leq \sqrt{2}\lambda^-$;

      iii. $b^2 c_{\text{new}} \eta_{\text{new}} \lambda^+_{\text{new}} \leq 0.5\lambda^-$ *(because* $b \leq 0.1$*, this will be satisfied if* $c_{\text{new}} \eta_{\text{new}} \lambda^+_{\text{new}} \leq 50\lambda^-$*);*

4. *The support of* $\boldsymbol{x}_t$ *changes enough so that for the* α *chosen above, Signal Model 3.2.11 holds with* $\beta = h^+\alpha$ *and*

$$\lceil \varrho \rceil^2 h^+ \leq 0.0025, \text{ and } \varrho_2 s\alpha \leq n$$

   or Signal Model 3.2.12 holds with $s \leq (6 \times 10^{-4})\alpha$ *and* $\alpha \leq \frac{n}{m}$.

5. *The low dimensional subspace is dense such that*

$$\kappa_{2s}(\boldsymbol{P}_{(J)}) \leq 0.3 \text{ and } \max_j \kappa_{2s}(\boldsymbol{P}_{(j),\text{new}}) \leq 0.02.$$

Then, with probability at least $1 - n^{-10}$*, at all times* t

1. *The support of* $\boldsymbol{x}_t$ *is recovered exactly, i.e.* $\hat{\mathcal{T}}_t = \mathcal{T}_t$;

2. *The estimate of the subspace change time satisfies* $t_j \leq \hat{t}_j \leq t_j + 2\alpha$*, for* $j = 1, \ldots, J$;

3. *The estimate of the number of new directions is correct, i.e.* $\hat{c}_{j,\text{new},k} = c_{j,\text{new}}$ *for* $j = 1, \ldots, J$ *and* $k = 1, \ldots, K$;

4. The recovery error satisfies:

$$\|\hat{\boldsymbol{x}}_t - \boldsymbol{x}_t\|_2 \leq \begin{cases} 1.2\left(\sqrt{\zeta} + \sqrt{c_{\text{new}}}\gamma_{\text{new}}\right) & t \in \left[t_j, \hat{t}_j\right] \\ 1.2\left(1.84\sqrt{\zeta} + (0.4)^{k-1}\sqrt{c_{\text{new}}}\gamma_{\text{new}}\right) & t \in \left[\hat{t}_j + (k-1)\alpha, \hat{t}_j + k\alpha - 1\right], \\ & k = 1, 2, \ldots, K \\ 2.4\sqrt{\zeta} & t \in \left[\hat{t}_j + K\alpha, t_{j+1} - 1\right]; \end{cases}$$

5. The subspace error $\text{SE}_t := \|(\boldsymbol{I} - \hat{\boldsymbol{P}}_t\hat{\boldsymbol{P}}_t')\boldsymbol{P}_t\|_2$ *satisfies:*

$$\text{SE}_t \leq \begin{cases} 1 & t \in \left[t_j, \hat{t}_j\right] \\ 10^{-2}\sqrt{\zeta} + 0.4^{k-1} & t \in \left[\hat{t}_j + (k-1)\alpha, \hat{t}_j + k\alpha - 1\right], \ k = 1, 2, \ldots, K \\ 10^{-2}\sqrt{\zeta} & t \in \left[\hat{t}_j + K\alpha, t_{j+1} - 1\right]. \end{cases}$$

Remark 3.2.16. *When $b = 0$ ($\boldsymbol{a}_t$'s are independent), the bound $\zeta \leq \frac{0.01\lambda^-}{b^2 r^3 \gamma^2}$ gets removed because $1/b^2 \to \infty$; and the third requirement in condition 3b) gets removed. Thus, when $b = 0$, the above theorem is the same as the main result in Chapter 2 even though it provides guarantees for a practical version of the ReProCS algorithm studied in Chapter 2.*

3.2.5 Random Support Change

We give here a commonly used Bernoulli-Gaussian motion model that satisfies our support change assumptions with high probability.

Model 3.2.17 (Taken from Chapter 2)**.** *Consider one-dimensional motion of the support of $\boldsymbol{x}_t$, and let $\tilde{o}_t$ be its center at time t. Suppose that the support moves according to the model*

$$o_t = o_{t-1} + \theta_t\left(1.1\frac{s}{\varrho} + \mu_t\right) \quad \text{and}$$

$$\tilde{o}_t = o_t \mod n$$

where μ_t is Gaussian $\mathcal{N}(0, \sigma^2)$ and θ_t is a Bernoulli random variable that takes the value 1 with probability q and 0 with probability $1 - q$, and $\varrho \geq 1$ is a constant. Taking the modulus with respect to n describes the process of the support starting over at 1 when it reaches n. Assume that $\{\mu_t\}$, $\{\theta_t\}$ are mutually independent and independent of $\{\boldsymbol{\nu}_t\}$ for $t = 1, \ldots, t_{\max}$.

Lemma 3.2.18 (Taken from Chapter 2)**.** *Signal Model 3.2.17 satisfies the assumptions of Signal Model 3.2.11 with* $\varrho_2 = \frac{1.2}{\varrho}$ *with probability at least* $1 - n^{-10}$ *if* $\sigma^2 \leq \dfrac{s^2}{4000\varrho^2 \log(n)}$, $q \geq 1 - \left(\dfrac{n^{-10}}{2(t_{\max} + \alpha)}\right)^{\frac{1}{\beta}}$, *and* $t_{\max} \leq n^{10}$.

Proof. To prove the above lemma, we will be done if we can show

1. the support changes at least once every β instants with probability greater than $1 - \frac{n^{-10}}{2}$;
2. when it changes, the support moves by at least $\frac{s}{\varrho}$ and not more than $1.2\frac{s}{\varrho}$ indices with probability greater than $1 - \frac{n^{-10}}{2}$.

Item 1) follows using simple arguments for Bernoulli random variables [24] while item 2) follows using a Gaussian tail bound. The complete proof is in Appendix 3.B. ⊠

Corollary 3.2.19. *Consider Theorem 3.2.15 with condition 4) replaced by Signal Model 3.2.17 with*

(i) $\dfrac{1.2s\alpha}{\varrho} \leq n$;

(ii) $t_{\max} \leq n^{10}$

(iii) $\sigma^2 \leq \dfrac{s^2}{4000\varrho^2 \log(n)}$;

(iv) $q \geq 1 - \left(\dfrac{n^{-10}}{2(t_{\max} + \alpha)}\right)^{\frac{1}{\beta}}$ *; for a* $\beta = \dfrac{(2.4\times10^{-3})\alpha}{\lceil\varrho\rceil^2}$.

Then all its conclusions will hold with probability greater than or equal to $1 - 2n^{-10}$.

Corollary 3.2.19 follows by combining Lemma 3.2.18 with Theorem 3.2.15.

3.2.6 Discussion

The above result needs an accurate estimate of the initial subspace, a slow subspace change assumption, a support change assumption, and a denseness assumption. If a short sequence of background only training data is available (which is often true of surveillance video), then the initial subspace estimate is easy to obtain by ordinary PCA. Otherwise, one could use a

batch method for robust PCA on an initial sequence to set $\hat{\boldsymbol{P}}_{(0)}$ before starting the ReProCS algorithm.

Consider the subspace change model. This model (along with the bound on γ_{new} from the theorem) assumes that after a subspace change, $\|\boldsymbol{a}_{t,\text{new}}\|_\infty$ and therefore also $\|\boldsymbol{\Lambda}_{a,t,\text{new}}\|_2$ are initially small. After t_j+d, $\|\boldsymbol{a}_{t,\text{new}}\|_\infty$ can be as large as $\gamma = \max_t \|\boldsymbol{a}_t\|_\infty$, and the eigenvalues of $\boldsymbol{\Lambda}_{a,t,\text{new}}$ can increase up to λ^+ either immediately or gradually. Thus a new direction added at time t_j can have magnitude as large as γ and variance as large as λ^+ by t_j+d. Since we assume that $t_j + d \le t_{j+1}$, this will occur before the next subspace change time. As demonstrated in [12], such a slow subspace change assumption is valid for backgrounds in real video sequences.

Consider the support change models. Both Models 3.2.11 and 3.2.12 are valid and commonly used models for foreground object motion in videos. Of course these are only special cases. Nothing prevents multiple moving objects (see Section 3.3.1). In the rest of this discussion we use Model 3.2.11. If we assume Model 3.2.11, our result requires $s \le \frac{n}{2\alpha}$. If $J \le C_1 n$ for some constant C_1, then using the definition of α, this bound holds if $s \le C_2 \frac{n}{\log n}$, for a constant C_2. Thus, this model allows $s \in \mathcal{O}(\frac{n}{\log n})$ and $r = r_0 + Jc_{\text{new}} \in \mathcal{O}(n)$. As we explain next, our denseness assumption restricts the requirement on r to $r \in \mathcal{O}(\log n)$.

Consider denseness. The way κ_s is defined, our denseness assumption simultaneously places restrictions on denseness of $\boldsymbol{\ell}_t$, and on r and s. As done in [2], we could assume $\kappa_1(\boldsymbol{P}_{(J)}) \le \sqrt{\frac{\mu r}{n}}$ and $\kappa_1(\boldsymbol{P}_{(j),\text{new}}) \le \sqrt{\frac{\tilde{\mu} c_{\text{new}}}{n}}$ where μ and $\tilde{\mu}$ take any value between 1 and $\frac{n}{r}$. It is easy to show that $\kappa_s(\boldsymbol{P}) \le \sqrt{s}\kappa_1(\boldsymbol{P})$ [12]. Thus if

$$\frac{2sr}{n} \le \mu^{-1}(0.3)^2, \text{ and } \frac{2sc_{\text{new}}}{n} \le \tilde{\mu}^{-1}(0.02)^2,$$

then our assumption of $\kappa_{2s}(\boldsymbol{P}_{(J)}) \le 0.3$ and $\kappa_{2s}(\boldsymbol{P}_{(j),\text{new}}) \le 0.02$ will be satisfied. Since we require $s \in \mathcal{O}(\frac{n}{\log n})$, this means we can allow $r \in \mathcal{O}(\log n)$ to satisfy the above.

Comparison with other work.

Let $\boldsymbol{L} = [\boldsymbol{\ell}_1, \boldsymbol{\ell}_2, \dots, \boldsymbol{\ell}_{t_{\max}}]$ and $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_{t_{\max}}]$. Define $r_{\text{mat}} := \text{rank}(\boldsymbol{L})$ and $s_{\text{mat}} := |\,\text{support}(\boldsymbol{X})|$. For our model, $s_{\text{mat}} = st_{\max}$, and $r_{\text{mat}} = r_J \le r$. From the above discussion, we see that the ReProCS result allows

$$s_{\text{mat}} \in \mathcal{O}\left(\frac{nt_{\max}}{\log n}\right) \text{ and } r_{\text{mat}} \in \mathcal{O}(\log n). \tag{3.5}$$

The above requirement on s_{mat} and r_{mat} is stronger than that used by [2] (which studies the batch approach PCP). The result in [2] allows

$$s_{\text{mat}} \in \mathcal{O}\left(nt_{\max}\right) \text{ and } r_{\text{mat}} \in \mathcal{O}\left(\frac{n}{(\log n)^2}\right).$$

But, up to differences in the constants, (3.5) is the same as the requirement found in [25] (which also studies the PCP program and is an improvement over [3]), except that [25] does not need specific bounds on s_{mat} and r_{mat}; it only requires $r_{\text{mat}} s_{\text{mat}} \in \mathcal{O}(nt_{\max})$. The comparison is not direct though because our result does not need denseness of the right singular vectors of $\boldsymbol{L}$ or a bound on the vector infinity norm of $\boldsymbol{UV'}$, while [2, 3], and [25] do. Here $\boldsymbol{L} \overset{\text{SVD}}{=} \boldsymbol{U\Sigma V'}$. The reason for our stronger requirement on $s_{\text{mat}} r_{\text{mat}}$ is because we study an online algorithm, ReProCS, that recovers the sparse vector $\boldsymbol{x}_t$ at each time t rather than in a batch or a piecewise batch fashion. Because of this, the sparse recovery step does not use the low dimensionality of the new (and still unestimated) subspace.

Because we only require that the support changes after a given maximum allowed duration, it can be constant for a certain period of time (Model 3.2.11), or it can change only a little at each time (Model 3.2.12). This is a substantially weaker assumption than the independent or uniformly random supports required by [2] and [20]. As we explain in Chapter 2, if we consider the whole matrix $\boldsymbol{X}$, then at most $\frac{t_{\max}}{5000}$ non-zero entries per row are allowed by our result. Thus, for $r_{\text{mat}} > 5000$, this also is a significant improvement over [25] which requires at most $\frac{t_{\max}}{r}$ non-zero entries per row. Therefore, an important advantage of our result is that it allows for highly correlated support sets of $\boldsymbol{x}_t$, which is important for applications such as video surveillance that involve one or more moving foreground objects or persons forming the sparse vector $\boldsymbol{x}_t$.

Now consider works that also use initial subspace knowledge. Our result improves upon [12]'s results by removing the denseness requirements on $(\boldsymbol{I} - \boldsymbol{P}_{(j),\text{new}}\boldsymbol{P}_{(j),\text{new}}{}')\hat{\boldsymbol{P}}_t$ and $(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}' - \hat{\boldsymbol{P}}_{t,\text{new}}\hat{\boldsymbol{P}}_{t,\text{new}}{}')\boldsymbol{P}_{(j),\text{new}}$ and thus provides a complete correctness result. It also improves on the results of Chapter 2 as explained earlier by studying a fully automatic version of ReProCS and assuming an autoregressive model on the $\boldsymbol{\ell}_t$'s. In [18], Feng et. al. propose a method for online robust PCA and prove a partial result for their algorithm. The approach is

to reformulate the PCP program and use this reformulation to develop a recursive algorithm that converges asymptotically to the solution of PCP as long as the basis estimate $\hat{\boldsymbol{P}}_t$ is full rank at each time t. Since this result assumes something about the algorithm estimates, it is only a *partial* result. Another work of Feng et. al. [19] on online robust PCA does not model the outlier as a sparse vector but defines anything that is far from the data subspace as an outlier. Another recent work that uses knowledge of the initial subspace estimate is modified-PCP [20]. However, like PCP, this also needs uniformly random supports. Moreover it is a piecewise batch approach.

Limitations. One limitation of this work is that we do not prove exact recovery of $\boldsymbol{x}_t$ or $\boldsymbol{\ell}_t$.

In order to set the algorithm parameters as assumed in Theorem 3.2.15 one would need knowledge of the model parameters γ, γ_{new}, r_0, c_{new}, λ^+, λ^-, and J. This is a rather impractical assumption; however, as shown below in Corollary 3.2.20, if an initial sequence of just $\boldsymbol{\ell}_t$'s is available, then this limitation can be handled with some additional assumptions.

Suppose that an initial sequence of just $\boldsymbol{\ell}_t$'s without any sparse corruptions is available. In the video surveillance application, this would correspond to having a short sequence of background only frames. Such a sequence can be used to obtain $\hat{\boldsymbol{P}}_{(0)}$ (as the eigenvectors corresponding to non-zero eigenvalues of $\sum_{t=1}^{t_{\text{train}}} \boldsymbol{\ell}_t \boldsymbol{\ell}_t{}'$) and to estimate several of the required model parameters. Let t_{train} be the length of the sequence and assume that $\boldsymbol{P}_t$ is constant for the duration. Define

$$\lambda^+ := \lambda_{\max}\left(\frac{1}{t_{\text{train}}}\sum_{t=1}^{t_{\text{train}}} \boldsymbol{\ell}_t \boldsymbol{\ell}_t{}'\right)$$

$$\lambda^- := \min_{\lambda_i \neq 0} \lambda_i\left(\frac{1}{t_{\text{train}}}\sum_{t=1}^{t_{\text{train}}} \boldsymbol{\ell}_t \boldsymbol{\ell}_t{}'\right)$$

$$r_0 := \operatorname{rank}\left(\sum_{t=1}^{t_{\text{train}}} \boldsymbol{\ell}_t \boldsymbol{\ell}_t{}'\right)$$

$$\gamma := \max_{t \in [1, t_{\text{train}}]} \|\boldsymbol{P}_t{}' \boldsymbol{\ell}_t\|_\infty.$$

Assume that

1. $c_{\text{new}} = 0.1 r_0$ and $\gamma_{\text{new}} = 0.1\gamma$

2. $\frac{\lambda_{\max}\left(\mathbf{\Lambda}_{\nu,t}\right)}{1-b^2} \leq \lambda^+$ for all t

3. $\frac{\lambda_{\min}\left(\mathbf{\Lambda}_{\nu,t}\right)}{1-b^2} \geq \lambda^-$ for all t

4. $c_{j,\text{new}} \leq c_{\text{new}} = 0.1 r_0$ for all j

5. $\frac{\|\boldsymbol{\nu}_t\|_\infty}{1-b} \leq \gamma$ for all t

6. $\frac{\|\boldsymbol{\nu}_{t,\text{new}}\|_\infty}{1-b} \leq \gamma_{\text{new}} = 0.1\gamma$ for $t \in [t_j, t_j + d]$ for all j.

Under these assumptions, the only remaining unknown parameter is J. In Theorem 3.2.15, knowledge of J is only used to control the probability with which the result holds. Therefore, we can state the following corollary.

Corollary 3.2.20. *Suppose that the above assumptions hold. In Theorem 3.2.15, replace the J in the expression for α with a 1, and suppose that all of the remaining assumptions of Theorem 3.2.15 are satisfied. Then all of the conclusions will also hold with probability at least $1 - Jn^{-10}$.*

Another limitation is that Signal Model 3.2.1 only allows for additions to the subspace. The more realistic model (Signal Model 3.2.8) also allows removals from the prior subspace. Nothing in our proof changes if we incorporate removals into the signal model, and we have the following corollary.

Corollary 3.2.21. *Theorem 3.2.15 also holds with Signal Model 3.2.1 on $\boldsymbol{\ell}_t$ replaced by Signal Model 3.2.8.*

Intuitively, Signal Model 3.2.8 is a special case of Signal Model 3.2.1 because Model 3.2.8 restricts the subspace where the $\boldsymbol{\ell}_t$ can lie. However, because of the minimum variance requirement (λ^-) of Model 3.2.1, we need technically need a separate proof. The proof remains the same; the only difference is that instead of estimating $\text{span}(\boldsymbol{P}_{(j)})$, the algorithm will maintain an estimate of $\text{span}([\boldsymbol{P}_{(0)}, \boldsymbol{P}_{(1),\text{new}}, \ldots, \boldsymbol{P}_{(j),\text{new}}])$. Notice that when there are no directions deleted, these are equivalent. Another way of describing this is that although directions are deleted from the subspace in Signal Model 3.2.8, Algorithm 3 does not delete anything from its subspace estimate. In Section 3.8 we introduce another algorithm (Algorithm 4) that does

remove old directions from its subspace estimate. A result similar to Theorem 3.2.15 is proven for this algorithm as well.

A fundamental limitation of our analysis approach is the assumption that there is a significant delay between times when the low-dimensional subspace changes. A more realistic model would allow the subspace to change more frequently.

Another limitation of the ReProCS algorithm is that it does not use the fact that the 'noise' seen by the sparse recovery step has a low dimensional structure. The modified PCP program from [20] uses this structure, but because of the piecewise batch approach cannot handle highly correlated supports of the sparse component like ReProCS (see the simulations is Section 3.10). The sparse recovery also requires a lower bound on $x_{\min}$ to recover the support.

3.3 Most General Support Change Model and a Key Lemma

We give here the most general support change model under which our result holds. We will show that this includes the preceding signal models as special cases.

3.3.1 Most General Support Change Model

The model given below is a simple generalization of the support change model used in Chapter 2.

Model 3.3.1. *Let ρ be a positive integer. Split $[1, t_{\max}]$ into intervals of length α. Use $\mathcal{J}_u := [(u-1)\alpha+1, u\alpha]$ to denote the u-th interval. For a given interval, $\mathcal{J}_u$, let $\mathcal{T}_{(i),u}$ for $i = 1, \dots, l_u$ be mutually disjoint subsets of $\{1, \dots, n\}$ such that for every $t \in \mathcal{J}_u$,*

$$\mathcal{T}_t \subseteq \mathcal{T}_{(i),u} \cup \mathcal{T}_{(i+1),u} \cup \cdots \cup \mathcal{T}_{(i+\rho-1),u} \quad \text{for some } i. \tag{3.6}$$

For these $\mathcal{T}_{(i),u}$'s define

$$h\left(\alpha; \{\mathcal{T}_{(i),u}\}_{\substack{u=1,\dots,\lceil \frac{t_{\max}}{\alpha} \rceil \\ i=1,\dots,l_u}}\right) :=$$

$$\max_{u=1,\dots,\lceil \frac{t_{\max}}{\alpha} \rceil} \max_i \left| \{t \in \mathcal{J}_u \ : \ \mathcal{T}_t \subseteq \mathcal{T}_{(i),u} \cup \mathcal{T}_{(i+1),u} \cup \cdots \cup \mathcal{T}_{(i+\rho-1),u}\} \right| \tag{3.7}$$

Now define $h^*(\alpha)$ *which takes the minimum over all choices of* $\mathcal{T}_{(i),u}$

$$h^*(\alpha) := \min_{\substack{\text{All choices of} \\ \text{mutually disjoint} \\ \mathcal{T}_{(i),u} \text{ satisfying (3.6)}}} h\left(\alpha; \{\mathcal{T}_{(i),u}\}_{\substack{u=1,\ldots,\lceil \frac{t_{\max}}{\alpha} \rceil \\ i=1,\ldots,l_u}}\right) \tag{3.8}$$

Assume that $h^*(\alpha) \leq h^+\alpha$ *for an* $h^+ < 1$.

In the above model, $h^*(\alpha)$ provides a bound on how long the support of $\boldsymbol{x}_t$ remains in a given area during an interval $\mathcal{J}_u$.

Notice that (3.6) can always be trivially satisfied by choosing $l_u = 1$ and $\mathcal{T}_{(1),u} = \{1, \ldots, n\}$, but this will give $h(\alpha; \{\mathcal{T}_{(i),u}\}) = \alpha$ and hence is not a good choice. This is why we take a minimum over all choices.

The following corollary is the most general form of our result.

Theorem 3.3.2. *Suppose that the conditions of Theorem 3.2.15 hold, but instead of 4), assume that Signal Model 3.3.1 holds with* $\rho^2 h^+ \leq .0024$. *Then all its conclusions will hold.*

In Sec. 3.3.3, we show that Signal Models 3.2.11 and 3.2.12 are special cases of Signal Model 3.3.1 and hence Theorem 3.2.15 is actually a corollary of Theorem 3.3.2. The rest of the paper will prove Theorem 3.3.2.

3.3.2 Key Lemma

Under Signal Model 3.3.1, we can obtain the following lemma for a matrix $\boldsymbol{M} = \sum_t \boldsymbol{I}_{\mathcal{T}_t} \boldsymbol{A}_t \boldsymbol{I}_{\mathcal{T}_t}{}'$ (i.e. a sum of matrices supported on rows and columns indexed by $\mathcal{T}_t$). This lemma tells us that, because of the support change model, such a matrix is actually block banded: for $\varrho = 1$ it is block diagonal, for $\varrho = 2$ it is block tri-diagonal, and so on. Hence its 2-norm is much smaller than the sum of the norms of the individual matrices. As we will see later, the error, $\boldsymbol{e}_t$, in recovering $\boldsymbol{x}_t$ (which is equal to the error in recovering $\boldsymbol{\ell}_t$) at times t, is supported on $\mathcal{T}_t$. As a result the matrix $\sum_t \boldsymbol{e}_t \boldsymbol{e}_t{}'$ has this form. Moreover, certain matrices obtained when bounding $\sum_t \boldsymbol{\ell}_t \boldsymbol{e}_t{}'$ also have this form.

Lemma 3.3.3 (Taken from Chapter 2)**.** *Let* $s_t = |\mathcal{T}_t|$. *Consider a sequence of* $s_t \times s_t$ *symmetric positive-semidefinite matrices* $\boldsymbol{A}_t$ *such that* $\|\boldsymbol{A}_t\|_2 \leq \sigma^+$ *for all* t. *Assume that the* $\mathcal{T}_t$ *obey Signal*

Model 3.3.1 and the assumptions of Theorem 3.3.2. Let $\boldsymbol{M} = \sum_{t \in \mathcal{J}_u} \boldsymbol{I}_{\mathcal{T}_t} \boldsymbol{A}_t \boldsymbol{I}_{\mathcal{T}_t}{}'$ *be an* $n \times n$ *matrix (*$\boldsymbol{I}$ *is an* $n \times n$ *identity matrix). Then*

$$\|\boldsymbol{M}\|_2 \leq \rho^2 h^+ \alpha \sigma^+ \leq 0.0024 \sigma^+ \alpha$$

Proof. This lemma is proved in Chapter 2. ⊠

3.3.3 Unifying the signal models

Lemma 3.3.4. *Suppose that Signal Model 3.2.11 and the conditions assumed in Theorem 3.2.15 hold. Recall that this means that the support of* $\boldsymbol{x}_t$ *is of a constant size* s *and moves by at least* $\frac{s}{\varrho}$ *indices, and at most* $\varrho_2 s$ *indices, at least every* β *time instants,* $\beta = h^+ \alpha$, $\lceil \varrho \rceil^2 h^+ \leq 0.0024$ *and* $\varrho_2 s \alpha \leq n$. *Then Signal Model 3.3.1 holds with* $\rho = \lceil \varrho \rceil$ *and* $h^+ = \beta / \alpha$.

The proof uses the same technique as in Chapter 2 and is given in Appendix 3.B, also see Figure 3.5.

Lemma 3.3.5. *Suppose that Signal Model 3.2.12 and the conditions assumed in Theorem 3.2.15 hold. Recall that this means that the support of* $\boldsymbol{x}_t$ *moves by between* 1 *and* m *indices at every time* t, $s \leq (3 \times 10^{-4}) \alpha$ *and* $\alpha \leq \frac{n}{m}$. *Then Signal Model 3.3.1 is satisfied with* $\rho = 2$ *and* $h^+ = s / \alpha$.

The proof is simple and is given in Appendix 3.B, also see Figure 3.5.

Remark 3.3.6. *In Signal Models 3.2.11 and 3.2.12, if we replace the assumption that the support restarts at 1 when it reaches* n, *and instead assume that it is reflected back up the vector, then a form of Lemma 3.3.3 still holds, but the conclusion becomes* $\|\boldsymbol{M}\|_2 \leq 2 \rho^2 h^+ \sigma^+ \alpha$. *The statement of Theorems 3.2.15 and 3.3.2 would then need the tighter bound on* $\rho^2 h^+ \leq .0012$.

In Lemma 3.2.18, we have already shown that Signal Model 3.2.17 is a special case of Signal Model 3.2.11. Above we have finished showing that both Signal Model 3.2.11 and 3.2.12 are special cases of Signal Model 3.3.1. In the rest of the paper, we assume Signal Model 3.3.1 and use Lemma 3.3.3 to prove Theorem 3.3.2. Theorem 3.2.15 follows as a special case of Theorem 3.3.2, which is the most general form of our result.

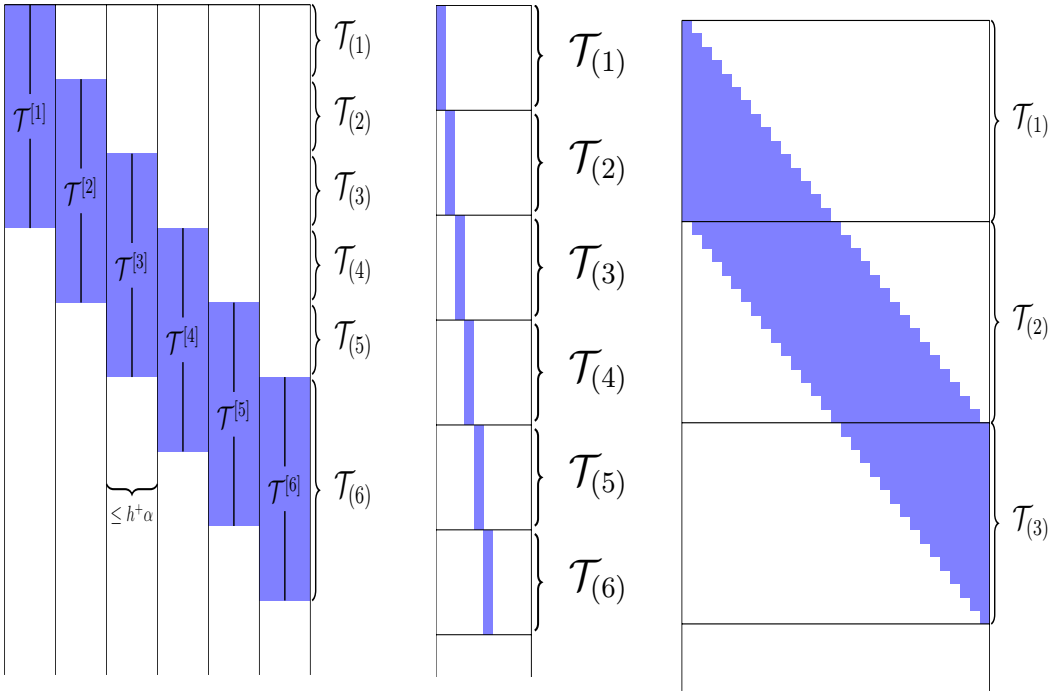

Figure 3.2 Signal Model 3.3.1 with $\varrho = 3$

Figure 3.3 Disjoint Supports (Signal Model 3.3.1 with $\varrho = 1$)

Figure 3.4 Signal Model 3.2.12

Figure 3.5: Support Change Models

3.4 The Automatic ReProCS Algorithm

The ReProCS algorithm was introduced in [12]. A more practical version including heuristics for setting the parameters was given in [14]. The basic idea of ReProCS is as follows. Given an accurate estimate of the subspace where the $\boldsymbol{\ell}_t$'s lie, projecting the measurement $\boldsymbol{m}_t = \boldsymbol{x}_t + \boldsymbol{\ell}_t$ onto the orthogonal complement of the estimated subspace will nullify most of $\boldsymbol{\ell}_t$. The denseness of $\boldsymbol{\ell}_t$ implies that this projection will have small RIC (Lemma 3.2.10) so the sparse recovery step will produce an accurate estimate $\hat{\boldsymbol{x}}_t$. Then, subtraction also gives a good estimate $\hat{\boldsymbol{\ell}}_t = \boldsymbol{m}_t - \hat{\boldsymbol{x}}_t$. Using these $\hat{\boldsymbol{\ell}}_t$, the algorithm successively updates the subspace estimate by a modification of the standard PCA procedure, which we call projection PCA.

Algorithm 3 is the same algorithm given in Chapter 2. The detailed description is given in Section 2.4

Algorithm 3 Recursive Projected CS (ReProCS) [12]

Parameters: ξ, ω, α, K, thresh
Input: $\boldsymbol{m}_t$, *Output*: $\hat{\boldsymbol{x}}_t$, $\hat{\boldsymbol{\ell}}_t$, $\hat{\boldsymbol{P}}_t$, $\hat{t}_j$, $\hat{c}_{j,\text{new},k}$
Notes: using $\hat{\boldsymbol{P}}_{t,*}, \hat{\boldsymbol{P}}_{t,\text{new}}, \hat{\boldsymbol{P}}_t = [\hat{\boldsymbol{P}}_{t,*}\ \hat{\boldsymbol{P}}_{t,\text{new}}]$
Set $\hat{\boldsymbol{P}}_{t,*} \leftarrow \hat{\boldsymbol{P}}_{(0)}$, $\hat{\boldsymbol{P}}_{t,\text{new}} \leftarrow [.]$, $j \leftarrow 0$, phase $\leftarrow$ detect
For every $t > 0$, do the following:

1. Estimate $\mathcal{T}_t$ and $\boldsymbol{x}_t$ via Projected Compressed Sensing:
   (a) Projection: set $\boldsymbol{\Phi}_t \leftarrow \boldsymbol{I} - \hat{\boldsymbol{P}}_{t-1}\hat{\boldsymbol{P}}_{t-1}'$, compute $\boldsymbol{y}_t \leftarrow \boldsymbol{\Phi}_t \boldsymbol{m}_t$
   (b) Sparse Recovery: compute $\hat{\boldsymbol{x}}_{t,\text{cs}}$ as the solution of $\min_{\boldsymbol{x}} \|\boldsymbol{x}\|_1$ $s.t.$ $\|\boldsymbol{y}_t - \boldsymbol{\Phi}_t \boldsymbol{x}\|_2 \le \xi$
   (c) Support Estimate: compute $\hat{\mathcal{T}}_t = \{i : \ |(\hat{\boldsymbol{x}}_{t,\text{cs}})_i| > \omega\}$
   (d) LS Estimate of $\boldsymbol{x}_t$: compute $(\hat{\boldsymbol{x}}_t)_{\hat{\mathcal{T}}_t} = ((\boldsymbol{\Phi}_t)_{\hat{\mathcal{T}}_t})^\dagger \boldsymbol{y}_t$, $(\hat{\boldsymbol{x}}_t)_{\hat{\mathcal{T}}_t^c} = 0$
2. Estimate $\boldsymbol{\ell}_t$: $\hat{\boldsymbol{\ell}}_t \leftarrow \boldsymbol{m}_t - \hat{\boldsymbol{x}}_t$.
3. If $t \mod \alpha \neq 0$ then $\hat{\boldsymbol{P}}_{t,*} \leftarrow \hat{\boldsymbol{P}}_{t-1,*}$, $\hat{\boldsymbol{P}}_{t,\text{new}} \leftarrow \hat{\boldsymbol{P}}_{t-1,\text{new}}$, $\hat{\boldsymbol{P}}_t \leftarrow [\hat{\boldsymbol{P}}_{t,*}\ \hat{\boldsymbol{P}}_{t,\text{new}}]$
4. If $t \mod \alpha = 0$ then detection or projection PCA
   If phase = detect then
   (a) Set $u = \frac{t}{\alpha}$ and compute $\boldsymbol{\mathcal{M}}_u = (\boldsymbol{I} - \hat{\boldsymbol{P}}_{u\alpha-1,*}\hat{\boldsymbol{P}}_{u\alpha-1,*}{}') \left(\frac{1}{\alpha}\sum_{\tau=(u-1)\alpha+1}^{u\alpha} \hat{\boldsymbol{\ell}}_\tau \hat{\boldsymbol{\ell}}_\tau{}'\right) (\boldsymbol{I} - \hat{\boldsymbol{P}}_{u\alpha-1,*}\hat{\boldsymbol{P}}_{u\alpha-1,*}{}')$
   (b) Update $\hat{\boldsymbol{P}}_{t,*} \leftarrow \hat{\boldsymbol{P}}_{t-1,*}$, $\hat{\boldsymbol{P}}_{t,\text{new}} \leftarrow \hat{\boldsymbol{P}}_{t-1,\text{new}}$, $\hat{\boldsymbol{P}}_t \leftarrow \hat{\boldsymbol{P}}_{t-1}$
   (c) If $\lambda_{\max}(\boldsymbol{\mathcal{M}}_u) \ge$ thresh then
      i. phase $\leftarrow$ ppca, increment $j \leftarrow j+1$, reset $k \leftarrow 0$
      ii. $\hat{u}_j \leftarrow u$, $\hat{t}_j = t$

   Else (phase = ppca) then
   (a) Set $u = \frac{t}{\alpha}$ and compute $\boldsymbol{\mathcal{M}}_u = (\boldsymbol{I} - \hat{\boldsymbol{P}}_{u\alpha-1,*}\hat{\boldsymbol{P}}_{u\alpha-1,*}{}') \left(\frac{1}{\alpha}\sum_{\tau=(u-1)\alpha+1}^{u\alpha} \hat{\boldsymbol{\ell}}_\tau \hat{\boldsymbol{\ell}}_\tau{}'\right) (\boldsymbol{I} - \hat{\boldsymbol{P}}_{u\alpha-1,*}\hat{\boldsymbol{P}}_{u\alpha-1,*}{}')$
   (b) Increment $k \leftarrow k+1$
   (c) Update $\hat{\boldsymbol{P}}_{t,\text{new}} \leftarrow$ eigenvectors$(\boldsymbol{\mathcal{M}}_u, \text{thresh})$, $\hat{\boldsymbol{P}}_{t,*} \leftarrow \hat{\boldsymbol{P}}_{t-1,*}$ and $\hat{\boldsymbol{P}}_t \leftarrow [\hat{\boldsymbol{P}}_{t,*}\ \hat{\boldsymbol{P}}_{t,\text{new}}]$. Set $\hat{c}_{j,\text{new},k} \leftarrow \text{rank}(\hat{\boldsymbol{P}}_{t,\text{new}})$.
   (d) If $k = K$
      i. Set phase $\leftarrow$ detect
      ii. Update $\hat{\boldsymbol{P}}_{t,*} \leftarrow \hat{\boldsymbol{P}}_t$, and $\hat{\boldsymbol{P}}_{t,\text{new}} \leftarrow [.]$

The function eigenvectors$(\boldsymbol{\mathcal{M}}, \text{thresh})$ returns a basis matrix for the span of all eigenvectors whose eigenvalue is above thresh.

3.5 Proof of Theorem 3.2.15

We will prove Theorem 3.3.2 which will imply Theorem 3.2.15.

3.5.1 Definitions

Definition 3.5.1. *Define*

$$\mathcal{J}_u := [(u-1)\alpha + 1, u\alpha].$$

Also define u_j to be the u such that $t_j \in \mathcal{J}_u$. That is $u_j := \left\lceil \frac{t_j}{\alpha} \right\rceil$. For the purposes of describing events before the first subspace change, let $u_0 := 0$.

Also define $\hat{u}_j := \frac{\hat{t}_j}{\alpha}$. Notice from the algorithm that this will be an integer, because detection only occurs when $t \mod \alpha = 0$.

We will show that with high probability, $\hat{u}_j = u_j$ or $\hat{u}_j = u_j + 1$.

Recall that

1. $\boldsymbol{P}_{(j),*} := \boldsymbol{P}_{(j-1)} = \boldsymbol{P}_{t_{j-1}}$
2. $\boldsymbol{P}_{(j),\text{new}} := \boldsymbol{P}_{t_j,\text{new}}$.

Definition 3.5.2. *For $j = 1, 2, \ldots, J$ and $k = 1, 2, \ldots, K$ define*

1. $\hat{\boldsymbol{P}}_{(j),*} := \hat{\boldsymbol{P}}_{\hat{t}_{j-1}+K\alpha}$. *This is the final estimate of $\boldsymbol{P}_{(j),*} = \boldsymbol{P}_{(j-1)}$.*
2. $\hat{\boldsymbol{P}}_{(j),\text{new},0} := [.]$ *and* $\hat{\boldsymbol{P}}_{(j),\text{new},k} := \hat{\boldsymbol{P}}_{\hat{t}_j+k\alpha,\text{new}}$ *is the k^{th} estimate of $\boldsymbol{P}_{(j),\text{new}}$.*

Notice from the algorithm that

1. $\hat{\boldsymbol{P}}_{t,*} = \hat{\boldsymbol{P}}_{(j),*}$ for all $t \in [\hat{t}_j, \hat{t}_j + K\alpha - 1]$
2. $\hat{\boldsymbol{P}}_{t,\text{new}} = \hat{\boldsymbol{P}}_{(j),\text{new},k}$ for all $t \in \mathcal{J}_{\hat{u}_j+(k+1)\alpha}$
3. At all times $\hat{\boldsymbol{P}}_t = [\hat{\boldsymbol{P}}_{t,*}\ \hat{\boldsymbol{P}}_{t,\text{new}}]$. Thus $\hat{\boldsymbol{P}}_t$ and $\hat{\boldsymbol{P}}_{t,\text{new}}$ update at every $t = \hat{t}_j + k\alpha$, $k = 1, 2, \ldots, K$, $j = 1, 2, \ldots, J$ while $\hat{\boldsymbol{P}}_{t,*}$ updates at every $t = \hat{t}_{j-1} + K\alpha$, $j = 2, \ldots, J$.
4. $\hat{\boldsymbol{P}}_{t-1,*} \perp \hat{\boldsymbol{P}}_{t,\text{new}}$
5. $\boldsymbol{\Phi}_t = (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}' - \hat{\boldsymbol{P}}_{(j),\text{new},k}\hat{\boldsymbol{P}}_{(j),\text{new},k}{}')$ when $t \in [\hat{u}_j + k\alpha + 1, \hat{u}_j + (k+1)\alpha]$

Definition 3.5.3. *Recall that for basis matrices* $\boldsymbol{P}$ *and* $\boldsymbol{Q}$, $\operatorname{dif}(\boldsymbol{P}, \boldsymbol{Q}) := \|(\boldsymbol{I} - \boldsymbol{P}\boldsymbol{P}')\boldsymbol{Q}\|_2$. *Define*

1. $\zeta_{j,*} := \operatorname{dif}(\hat{\boldsymbol{P}}_{(j),*}, \boldsymbol{P}_{(j),*})$
2. $\zeta_{j,\text{new},k} := \operatorname{dif}([\hat{\boldsymbol{P}}_{(j),*}\ \hat{\boldsymbol{P}}_{(j),\text{new},k}], \boldsymbol{P}_{(j),\text{new}})$

Recall $\text{SE}_t = \operatorname{dif}(\hat{\boldsymbol{P}}_t, \boldsymbol{P}_t)$, *and notice that for* $t \in \mathcal{J}_{\hat{u}+k}$, $\text{SE}_t \leq \zeta_{j,*} + \zeta_{j,\text{new},k-1}$.

Definition 3.5.4. *Define*

1. $\zeta^+_{j,*} := \big(r_0 + (j-1)c_{\text{new}}\big)\zeta$
2. $\zeta^+_{0,\text{new}} := 1$, $\zeta^+_{k,\text{new}} := \dfrac{b_{\boldsymbol{\mathcal{H}},k}}{b_{\boldsymbol{A}} - b_{\boldsymbol{A},\perp} - b_{\boldsymbol{\mathcal{H}},k}}$ *(the right hand side depends on* $\zeta^+_{k-1,\text{new}}$*)*

where $b_{\boldsymbol{A}}$, $b_{\boldsymbol{A},\perp}$, *and* $b_{\boldsymbol{\mathcal{H}},k}$ *are defined in* (3.11), (3.12), *and* (3.13), *respectively.*

Definition 3.5.5. *Define the random variable*

$$X_u := \{\boldsymbol{\nu}_1, \ldots, \boldsymbol{\nu}_{u\alpha}\}$$

If assuming Signal Model 3.2.17 (random support change), then both $\{\theta_t\}$ *and* $\{\mu_t\}$ *for* $t = 1, \ldots, t_{\max}$ *are also included in the definition of* X_u *for all* u. *Thus whenever conditioning on an* X_u, *they can be treated as deterministic.*

Definition 3.5.6. *For* $j = 1, \ldots, J$, $k = 1, \ldots, K$, *and for* $a = u_j$ *or* $a = u_j + 1$, *define the following events*

- $\text{DET}^a_j := \{\hat{u}_j = a\}$
- $\text{PPCA}^a_{j,k} := \Big\{\hat{u}_j = a \text{ and } \operatorname{rank}(\hat{\boldsymbol{P}}_{(j),\text{new},k}) = c_{j,\text{new}} \text{ and } \zeta_{j,\text{new},k} \leq \zeta^+_{k,\text{new}}\Big\}$
- $\text{NODETS}^a_j := \{\hat{u}_j = a \text{ and } \lambda_{\max}(\boldsymbol{\mathcal{M}}_u) < \text{thresh for all } u \in [\hat{u}_j + K + 1, u_{j+1} - 1]\}$
- $\Gamma^a_{j,0} := \Gamma_{j-1,\text{end}} \cap \text{DET}^a_j$
- $\Gamma^a_{j,k} := \Gamma^a_{j,k-1} \cap \text{PPCA}^a_{j,k}$
- $\Gamma_{j,\text{end}} := \Big(\Gamma^{u_j}_{j,K} \cap \text{NODETS}^{u_j}_j\Big) \cup \Big(\Gamma^{u_j+1}_{j,K} \cap \text{NODETS}^{u_j+1}_j\Big)$

- $\Gamma_{0,\text{end}} := \{\zeta_{1,*} \leq r_0\zeta\} \cap \{\lambda_{\max}(\boldsymbol{\mathcal{M}}_u) < \text{thresh for all } u \in [1, u_1 - 1]\}$

We misuse notation as follows. Suppose that a set Γ is a subset of all possible values that a r.v. X can take. For two r.v.s' $\{X, Y\}$, when we need to say "$X \in \Gamma$ and Y can be anything" we will sometimes misuse notation and just say "$\{X, Y\} \in \Gamma$." For example, we sometimes say $X_{u_j} \in \Gamma_{j,\text{end}}$. This means $X_{u_j-1} \in \Gamma_{j,\text{end}}$ and $\boldsymbol{\nu}_t$ for $t \in \mathcal{J}_{u_j}$ are unconstrained.

Definition 3.5.7. *Define $\boldsymbol{e}_t$ to be the error made in estimating $\boldsymbol{x}_t$ and $\boldsymbol{\ell}_t$. That is*

$$\boldsymbol{e}_t := \hat{\boldsymbol{x}}_t - \boldsymbol{x}_t = \boldsymbol{\ell}_t - \hat{\boldsymbol{\ell}}_t$$

3.5.2 Main Lemmas

Fact 3.5.8. *Under the assumptions of Theorem 3.2.15, $\frac{1}{\alpha} \leq (c_{\text{new}}\zeta)^2$. To see this, observe that the lower bound for α has $(c_{\text{new}}\zeta)^2$ in the denominator, and everything else in the expression is greater than or equal to 1. (Notice that $\frac{{\gamma_{\text{new}}}^2}{\lambda^-} \geq 1$)*

Lemma 3.5.9. *Under the conditions of Theorem 3.2.15,*

$$\zeta^+_{k,\text{new}} \leq 0.4^k + 0.84 c_{\text{new}}\zeta$$

Proof. This claim follows by applying simple algebra on the definition and using the bounds assumed on α, ζ, and b in Theorem 3.2.15. In particular, we use the fact that $\frac{1}{\alpha} \leq (c_{\text{new}}\zeta)^2$ (Fact 3.5.8). Detailed proofs of similar results can be found in [12] and [13, Lemma 6.2]. Those proofs define the quantity $g := \frac{\lambda^+_{\text{new}}}{\lambda^-_{\text{new}}}$ and assume $g \leq \sqrt{2}$. Although we do not explicitly define g, our assumption that $\lambda^+_{\text{new}} \leq \sqrt{2}\lambda^-$ implies that $\frac{\lambda^+_{\text{new}}}{\lambda^-_{\text{new}}} \leq \sqrt{2}$, because $\lambda^-_{\text{new}} \geq \lambda^-$.

For the purposes of review, we have included a proof of this lemma in Appendix 3.C. ⊠

Lemma 3.5.10 (Sparse Recovery Lemma (Chapter 2, Lemma 2.6.15))**.** *Assume that all of the conditions of Theorem 3.2.15 hold. Recall that* $\text{SE}_t = \text{dif}(\hat{\boldsymbol{P}}_t, \boldsymbol{P}_t)$.

1. *Conditioned on $\Gamma_{j-1,\text{end}}$, for $t \in [t_j, (u_j + 1)\alpha]$*

   (a) $\phi_t := \|[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\|_2 \leq \phi^+ := 1.2$.

(b) the support of $\boldsymbol{x}_t$ *is recovered exactly i.e.* $\hat{\mathcal{T}}_t = \mathcal{T}_t$ *and* $\boldsymbol{e}_t$ *satisfies:*

$$\boldsymbol{e}_t := \hat{\boldsymbol{x}}_t - \boldsymbol{x}_t = \boldsymbol{\ell}_t - \hat{\boldsymbol{\ell}}_t = \boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{\Phi}_t\boldsymbol{\ell}_t. \tag{3.9}$$

(c) Furthermore,

$$\mathrm{SE}_t \le 1 \ , \ \textit{and}$$
$$\|\boldsymbol{e}_t\|_2 \le \phi^+(\zeta^+_{j,*}\sqrt{r}\gamma + \sqrt{c_{\mathrm{new}}}\gamma_{\mathrm{new}}) \le 1.2\left(\sqrt{\zeta} + \sqrt{c_{\mathrm{new}}}\gamma_{\mathrm{new}}\right)$$

2. *For* $k = 1, 2, \ldots, K$ *and* $\hat{u}_j = u_j$ *or* $\hat{u}_j = u_j + 1$, *conditioned on* $\Gamma^{\hat{u}_j}_{j,k-1}$, *for* $t \in [(\hat{u}_j + k - 1)\alpha + 1, (\hat{u}_j + k)\alpha]$, *the first two conclusions above hold. That is,* $\phi_t \le \phi^+$ *and* $\boldsymbol{e}_t$ *satisfies (3.9). Furthermore,*

$$\mathrm{SE}_t \le \zeta^+_{j,*} + \zeta^+_{j,\mathrm{new},k-1} \ , \ \textit{and}$$
$$\|\boldsymbol{e}_t\|_2 \le \phi^+(\zeta^+_{j,*}\sqrt{r}\gamma + \zeta^+_{j,\mathrm{new},k-1}\sqrt{c_{\mathrm{new}}}\gamma_{\mathrm{new}}) \le 1.2\left(1.84\sqrt{\zeta} + (0.4)^{k-1}\sqrt{c_{\mathrm{new}}}\gamma_{\mathrm{new}}\right)$$

3. *For* $\hat{u}_j = u_j$ *or* $\hat{u}_j = u_j + 1$, *conditioned on* $\Gamma^{\hat{u}_j}_{j,K}$, *for* $t \in [(\hat{u}_j + K)\alpha + 1, t_{j+1} - 1]$, *the first two conclusions above hold (*$\phi_t \le \phi^+$ *and* $\boldsymbol{e}_t$ *satisfies* (3.9)*). Furthermore,*

$$\mathrm{SE}_t \le \zeta^+_{j+1,*} \ , \ \textit{and}$$
$$\|\boldsymbol{e}_t\|_2 \le \phi^+\zeta^+_{j+1,*}\sqrt{r}\gamma \le 1.2\sqrt{\zeta}$$

The proof in Chapter 2 uses the facts that under the various conditionings, $\mathrm{rank}(\hat{\boldsymbol{P}}_{(j),k-1,\mathrm{new}}) = c_{j,\mathrm{new}}$, $\zeta_{j,*} \le \zeta^+_{j,*}$, and $\zeta_{j,\mathrm{new},k-1} \le \zeta^+_{k-1,\mathrm{new}}$.

Lemma 3.5.11.

1. *The event* $\Gamma^{\hat{u}_j}_{j,K}$ *and so also the event* $\Gamma_{j,\mathrm{end}}$ *imply that* $\zeta_{j+1,*} \le \zeta^+_{j+1,*}$.

2. $\mathbb{P}\left(\mathrm{NODETS}^a_j \mid \Gamma^a_{j,K}\right) = 1$ *for* $a = u_j$ *or* $a = u_j + 1$.

Lemma 3.5.12. *For* $j = 1, \ldots, J$,

$$\mathbb{P}\left(\mathrm{DET}^{u_j+1} \mid \Gamma_{j-1,\mathrm{end}}, \overline{\mathrm{DET}^{u_j}}\right) \ge p_{\mathrm{det},1} := 1 - p_{\boldsymbol{A}} - p_{\boldsymbol{\mathcal{H}}}.$$

The definitions of $p_{\boldsymbol{A}}$ *and* $p_{\boldsymbol{\mathcal{H}}}$ *can be found in Lemmas 3.5.20 and 3.5.22 respectively.*

Fact 3.5.13. *Observe that* $\Gamma^a_{j,0}$ *for* $a = u_j$ *or* $a = u_j + 1$ *implies that* $u_j \le \hat{u}_j \le u_j + 1$. *Thus,* $t_j \le \hat{t}_j \le t_j + 2\alpha$. *So with the model assumption that* $d \ge (K+2)\alpha$, *we have that* $\mathcal{J}_{\hat{u}_j+k} \subseteq [t_j, t_j + d]$ *for* $k = 1, 2, \ldots, K$. *This fact is needed so that the tighter bounds on* $\boldsymbol{a}_{t,\text{new}}$ *hold from condition 3b in the Theorem.*

Lemma 3.5.14.

$$\mathbb{P}\left(\Gamma^a_{j,k} \mid \Gamma^a_{j,k-1}\right) = \mathbb{P}\left(\text{PPCA}^a_{j,k} \mid \Gamma^a_{j,k-1}\right) \ge p_{\text{ppca}} := 1 - p_{\boldsymbol{A}} - p_{\boldsymbol{A},\perp} - p_{\boldsymbol{\mathcal{H}}}$$

for $a = u_j$ *or* $a = u_j + 1$. *where* $p_{\text{ppca}} := 1 - p_{\boldsymbol{A}} - p_{\boldsymbol{A},\perp} - p_{\boldsymbol{\mathcal{H}}}$. *The definitions of* $p_{\boldsymbol{A}}$, $p_{\boldsymbol{A},\perp}$, *and* $p_{\boldsymbol{\mathcal{H}}}$ *can be found in Lemmas 3.5.20, 3.5.21, and 3.5.22 respectively.*

The above lemma says that whether the new directions are detected at u_j or $u_j + 1$, conditioned on $k-1$ previous successful projection PCA steps, the probability of a successful k^{th} projection PCA step is lower bounded by p_{ppca}.

Corollary 3.5.15. *Let*

$$p_{\text{det},0} := \mathbb{P}\left(\text{DET}^{u_j} \mid \Gamma_{j-1,\text{end}}\right)$$

and therefore, $1 - p_{\text{det},0} = \mathbb{P}\left(\overline{\text{DET}^{u_j}} \mid \Gamma_{j-1,\text{end}}\right)$.

From the above lemmas, we get that

$$\begin{aligned}
\mathbb{P}\left(\Gamma_{j,\text{end}} \mid \Gamma_{j-1,\text{end}}\right) &= \mathbb{P}\Big(\left(\text{DET}^{u_j} \cap \text{PPCA}^{u_j}_{j,1} \cap \cdots \cap \text{PPCA}^{u_j}_{j,K} \cap \text{NODETS}^{u_j}_{j}\right) \cup \\
&\qquad \left(\overline{\text{DET}^{u_j}} \cap \text{DET}^{u_j+1} \cap \text{PPCA}^{u_j+1}_{j,k} \cap \cdots \cap \text{PPCA}^{u_j+1}_{j,K} \cap \text{NODETS}^{u_j+1}_{j}\right) \mid \Gamma_{j-1,\text{end}}\Big) \\
&= \mathbb{P}\left(\text{DET}^{u_j} \cap \text{PPCA}^{u_j}_{j,1} \cap \cdots \cap \text{PPCA}^{u_j}_{j,K} \mid \Gamma_{j-1,\text{end}}\right) \\
&\quad + \mathbb{P}\left(\overline{\text{DET}^{u_j}} \cap \text{DET}^{u_j+1} \cap \text{PPCA}^{u_j+1}_{j,k} \cap \cdots \cap \text{PPCA}^{u_j+1}_{j,K} \mid \Gamma_{j-1,\text{end}}\right) \\
&\ge p_{\text{det},0} \cdot (p_{\text{ppca}})^K + (1 - p_{\text{det},0}) \cdot p_{\text{det},1} \cdot (p_{\text{ppca}})^K \\
&\ge p_{\text{det},0} \cdot p_{\text{det},1} \cdot (p_{\text{ppca}})^K + (1 - p_{\text{det},0}) \cdot p_{\text{det},1} \cdot (p_{\text{ppca}})^K \\
&= p_{\text{det},1} \cdot (p_{\text{ppca}})^K.
\end{aligned}$$

Proof of Theorem 3.3.2. Theorem 3.3.2 follows from Corollary 3.5.15 and the assumed lower bound on α. Notice that by Lemma 3.5.9, the choice of K, and Lemma 3.5.10 , the event $\Gamma_{J,\text{end}}$ will imply all conclusions of the theorem.

By the first assumption of Theorem 3.2.15 and the argument used to prove Lemma 3.5.11, we get that $\mathbb{P}(\Gamma_{0,\text{end}}) = 1$.

Next, we have by the chain rule,

$$\mathbb{P}(\Gamma_{J,\text{end}}) = \prod_{j=1}^{J} \mathbb{P}(\Gamma_{j,\text{end}} \mid \Gamma_{j-1,\text{end}}, \Gamma_{j-2,\text{end}}, \ldots, \Gamma_{1,\text{end}}, \Gamma_{0,\text{end}}).$$

Because $\Gamma_{j-1,\text{end}} \subseteq \Gamma_{j-2,\text{end}} \subseteq \cdots \subseteq \Gamma_{1,\text{end}} \subseteq \Gamma_{0,\text{end}}$, we get

$$\begin{aligned}
\mathbb{P}(\Gamma_{J,\text{end}}) &= \prod_{j=1}^{J} \mathbb{P}(\Gamma_{j,\text{end}} \mid \Gamma_{j-1,\text{end}}) \\
&\geq \prod_{j=1}^{J} p_{\text{det},1} \cdot (p_{\text{ppca}})^K \\
&= (p_{\text{det},1} \cdot (p_{\text{ppca}})^K)^J \\
&\geq 1 - n^{-10}
\end{aligned}$$

The last line is by the lower bound on α assumed in Theorem 3.2.15. ⊠

3.5.3 Key Lemmas for Proving of Lemmas 3.5.11, 3.5.12, and 3.5.14

Before proving the lemmas from the preceding section, we introduce several lemmas which will be used in the proofs. Their statement requires the following definition.

Definition 3.5.16. *Recall from Algorithm 3 that* $\boldsymbol{\mathcal{M}}_u = \frac{1}{\alpha} \sum_{t \in \mathcal{J}_u} (\boldsymbol{I} - \hat{\boldsymbol{P}}_{u\alpha-1,*} \hat{\boldsymbol{P}}_{u\alpha-1,*}{}') \hat{\boldsymbol{\ell}}_t \hat{\boldsymbol{\ell}}_t{}' (\boldsymbol{I} - \hat{\boldsymbol{P}}_{u\alpha-1,*} \hat{\boldsymbol{P}}_{u\alpha-1,*}{}')$. *Using the definition of* $\boldsymbol{P}_{(j),*}$, *for* $u = u_j + 1$ *or* $u = \hat{u}_j + k$,

$$\boldsymbol{\mathcal{M}}_u = (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*} \hat{\boldsymbol{P}}_{(j),*}{}') \left(\frac{1}{\alpha} \sum_{t \in \mathcal{J}_u} \hat{\boldsymbol{\ell}}_t \hat{\boldsymbol{\ell}}_t{}' \right) (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*} \hat{\boldsymbol{P}}_{(j),*}{}').$$

Define

1. *Let* $\boldsymbol{D}_{j,\text{new}} := (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*} \hat{\boldsymbol{P}}_{(j),*}{}') \boldsymbol{P}_{(j),\text{new}} \overset{QR}{=} \boldsymbol{E}_{j,\text{new}} \boldsymbol{R}_{j,\text{new}}$ *denote its reduced QR decomposition, i.e. let* $\boldsymbol{E}_{j,\text{new}}$ *be a basis matrix for* $\text{range}\,(\boldsymbol{D}_{j,\text{new}})$ *and let* $\boldsymbol{R}_{j,\text{new}} = \boldsymbol{E}_{j,\text{new}}{}' \boldsymbol{D}_{j,\text{new}}$.

2. *Let* $\boldsymbol{E}_{j,\text{new},\perp}$ *be a basis matrix for the orthogonal complement of* $\text{range}(\boldsymbol{E}_{j,\text{new}})$. *To be precise,* $\boldsymbol{E}_{j,\text{new},\perp}$ *is an* $n \times (n - r_j)$ *basis matrix so that* $[\boldsymbol{E}_{j,\text{new}}\ \boldsymbol{E}_{j,\text{new},\perp}]$ *is unitary.*

3. For $u = u_j + 1$ and $u = \hat{u}_j + k$ for $k = 1, \ldots, K$, define $\boldsymbol{A}_u$, $\boldsymbol{A}_{u,\perp}$, $\boldsymbol{\mathcal{A}}_u$ and $\boldsymbol{\mathcal{H}}_u$ as

$$\boldsymbol{A}_u := \frac{1}{\alpha} \sum_{t \in \mathcal{J}_u} \boldsymbol{E}_{j,\text{new}}{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{\ell}_t\boldsymbol{\ell}_t{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{E}_{j,\text{new}}$$

$$\boldsymbol{A}_{u,\perp} := \frac{1}{\alpha} \sum_{t \in \mathcal{J}_u} \boldsymbol{E}_{j,\text{new},\perp}{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{\ell}_t\boldsymbol{\ell}_t{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{E}_{j,\text{new},\perp}$$

and let

$$\boldsymbol{\mathcal{A}}_u := \begin{bmatrix} \boldsymbol{E}_{j,\text{new}} & \boldsymbol{E}_{j,\text{new},\perp} \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_u & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A}_{u,\perp} \end{bmatrix} \begin{bmatrix} \boldsymbol{E}_{j,\text{new}}{}' \\ \boldsymbol{E}_{j,\text{new},\perp}{}' \end{bmatrix}$$

and

$$\boldsymbol{\mathcal{H}}_u := \boldsymbol{\mathcal{M}}_u - \boldsymbol{\mathcal{A}}_u$$

Recall that for the above values of u_j,

$$\boldsymbol{\mathcal{M}}_u = (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}') \left(\frac{1}{\alpha} \sum_{t \in \mathcal{J}_u} \hat{\boldsymbol{\ell}}_t\hat{\boldsymbol{\ell}}_t{}' \right) (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}').$$

When $u = \hat{u}_j + k$ for a $k \leq K$, $\boldsymbol{\mathcal{M}}_u$ is the matrix whose eigenvectors with eigenvalue above thresh *form $\hat{\boldsymbol{P}}_{(j),\text{new},k}$ (see step 4c of Algorithm 3). In this case, $\boldsymbol{\mathcal{M}}_u$ has eigendecomposition*

$$\boldsymbol{\mathcal{M}}_u \overset{\text{EVD}}{=} \begin{bmatrix} \hat{\boldsymbol{P}}_{(j),\text{new},k} & \hat{\boldsymbol{P}}_{(j),k,\text{new},\perp} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\Lambda}}_u & \boldsymbol{0} \\ \boldsymbol{0} & \hat{\boldsymbol{\Lambda}}_{u,\perp} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{P}}_{(j),\text{new},k}{}' \\ \hat{\boldsymbol{P}}_{(j),k,\text{new},\perp}{}' \end{bmatrix}.$$

Note: we use $\boldsymbol{A}_u$, $\boldsymbol{A}_{u,\perp}$, and $\boldsymbol{\mathcal{H}}_u$ for $u = u_j + 1$ or for $u = \hat{u}_j + 1, \ldots, \hat{u}_j + K$. With the appropriate conditioning, all these u's lie in the interval $[u_j + 1, u_j + d - 1]$ and from the assumptions in the Theorem, in this interval $\boldsymbol{a}_{t,\text{new}}$ is "small".

The following lemma follows from the $\sin\theta$ Theorem [26] and Weyl's theorem. It is taken from [12].

Lemma 3.5.17 ([12])**.** *At $u = \hat{u}_j + k$, if $\text{rank}(\hat{\boldsymbol{P}}_{(j),\text{new},k}) = c_{j,\text{new}}$, and if $\lambda_{\min}(\boldsymbol{A}_u) - \|\boldsymbol{A}_{u,\perp}\|_2 - \|\boldsymbol{\mathcal{H}}_u\|_2 > 0$, then*

$$\zeta_{j,\text{new},k} \leq \frac{\|\boldsymbol{\mathcal{H}}_u\|_2}{\lambda_{\min}(\boldsymbol{A}_u) - \|\boldsymbol{A}_{u,\perp}\|_2 - \|\boldsymbol{\mathcal{H}}_u\|_2} \tag{3.10}$$

The next three lemmas each assert a high probability bound for one of the terms in (3.10).

Definition 3.5.18. *For ease of notation, define the function*

$$F(\alpha, \epsilon, b_1) := \exp\left(\frac{-\alpha\epsilon^2}{8(b_1)^2}\right)$$

In the following lemmas, let

$$\epsilon = 0.01 c_{\text{new}} \zeta \lambda^-.$$

Remark 3.5.19. *In the next three lemmas we use quantities $b_{\boldsymbol{A}}$, $b_{\boldsymbol{A},\perp}$, and $b_{\boldsymbol{\mathcal{H}},k}$. These are not to be confused with the autoregression (AR) coefficient b used for the AR model on $\boldsymbol{a}_t$ (see Signal Model 3.2.1).*

Let

$$p_{\boldsymbol{A}} := cF(\alpha, \epsilon, c_{\text{new}}{\gamma_{\text{new}}}^2) + 3(r + c_{\text{new}})F(\alpha, \epsilon, 2\sqrt{c_{\text{new}} r}\gamma\gamma_{\text{new}}) + (r + c_{\text{new}})F(\alpha, \epsilon, 4\sqrt{c_{\text{new}} r}\gamma\gamma_{\text{new}})$$

and

$$b_{\boldsymbol{A}} := (1 - (\zeta_{j,*}^+)^2)(1 - b^2)\lambda_{\text{new}}^- - \epsilon - 2\zeta_{j,*}^+ \left(\frac{b^2}{\alpha(1-b^2)}\sqrt{c_{\text{new}} r}\gamma_{\text{new}}\gamma + 4\epsilon\right). \tag{3.11}$$

Lemma 3.5.20. *For $k = 1, \ldots, K$,*

$$\mathbb{P}\left(\lambda_{\min}\left(\boldsymbol{A}_{\hat{u}_j+k}\right) \geq b_{\boldsymbol{A}} \;\middle|\; X_{\hat{u}_j+k-1}\right) \geq 1 - p_{\boldsymbol{A}}$$

for all $X_{\hat{u}_j+k-1} \in \Gamma_{j,k-1}^{\hat{u}_j}$ with $\hat{u}_j = u_j$ or $\hat{u}_j = u_j + 1$. Also,

$$\mathbb{P}\left(\lambda_{\min}\left(\boldsymbol{A}_{u_j+1}\right) \geq b_{\boldsymbol{A}} \;\middle|\; X_{u_j}\right) \geq 1 - p_{\boldsymbol{A}}$$

for all $X_{u_j} \in \Gamma_{j-1,\text{end}}$.

Let $p_{\boldsymbol{A},\perp} := rF\left(\alpha, \epsilon, (\zeta_{j,*}^+)^2 r\gamma^2\right)$ and

$$b_{\boldsymbol{A},\perp} := (\zeta_{j,*}^+)^2(b^2 r\gamma^2 + (1-b^2)\lambda^+ + \epsilon). \tag{3.12}$$

Lemma 3.5.21. *For $k = 1, \ldots, K$,*

$$\mathbb{P}\left(\lambda_{\max}\left(\boldsymbol{A}_{\hat{u}_j+k,\perp}\right) \leq b_{\boldsymbol{A},\perp} \;\middle|\; X_{\hat{u}_j+k-1}\right) \geq 1 - p_{\boldsymbol{A},\perp}$$

for all $X_{\hat{u}_j+k-1} \in \Gamma_{j,k-1}^{\hat{u}_j}$ with $\hat{u}_j = u_j$ or $\hat{u}_j = u_j + 1$.

The same bound holds for $\|\boldsymbol{A}_{u_j+1,\perp}\|_2$ when we condition on $X_{u_j} \in \Gamma_{j-1,\text{end}}$.

Let $p_{\boldsymbol{\mathcal{H}}} := nF\left(\alpha, \epsilon, \left[\phi^+\left(\zeta_{j,*}^+\sqrt{r}\gamma + \sqrt{c_{\text{new}}}\gamma_{\text{new}}\right)\right]^2\right) + rF\left(\alpha, \epsilon, r\gamma^2\right) + c_{\text{new}}F\left(\alpha, \epsilon, {c_{\text{new}}\gamma_{\text{new}}}^2\right) + 3(r + c_{\text{new}})F(\alpha, \epsilon, 2\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}) + (r + c_{\text{new}})F(\alpha, \epsilon, 4\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}) + c_{\text{new}}F\left(\alpha, \epsilon, r\gamma^2\right) + 3(r + c_{\text{new}})F(\alpha, \epsilon, 2\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}) + (r + c_{\text{new}})F(\alpha, \epsilon, 4\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}})$. Define $\kappa_s^+ := .0215$. Also let

$$b_{\boldsymbol{\mathcal{H}},k} := b_{2,k} + 2b_{4,k} + 2b_6, \tag{3.13}$$

where

$$b_{2,k} := \begin{cases} \rho^2 h^+(\phi^+)^2(\zeta_{j,*}^+)^2(b^2 r\gamma^2 + (1-b^2)\lambda^+) + \\ \rho^2 h^+(\phi^+)^2(\kappa_s^+)^2(b^2 {c_{\text{new}}\gamma_{\text{new}}}^2 + (1-b^2)\lambda_{\text{new}}^+) + & k = 1 \\ 2\cdot\rho^2 h^+(\phi^+)^2\kappa_s^+\zeta_{j,*}^+ b^2\sqrt{rc_{\text{new}}}\gamma\gamma_{\text{new}} \\ \\ \rho^2 h^+(\phi^+)^2(\zeta_{j,*}^+)^2(b^2 r\gamma^2 + (1-b^2)\lambda^+) + \\ \rho^2 h^+(\phi^+)^2(\zeta_{j,\text{new},k-1}^+)^2(b^2 {c_{\text{new}}\gamma_{\text{new}}}^2 + (1-b^2)\lambda_{\text{new}}^+) + & k \geq 2 \\ 2\cdot\rho^2 h^+(\phi^+)^2\zeta_{j,*}^+\zeta_{j,\text{new},k-1}^+ b^2\sqrt{rc_{\text{new}}}\gamma\gamma_{\text{new}} \end{cases}$$

$$b_{4,k} := \begin{cases} \Big[(\zeta_{j,*}^+)^2(b^2 r\gamma^2 + (1-b^2)\lambda^+ + \epsilon) + \\ \kappa_s^+(b^2 {c_{\text{new}}\gamma_{\text{new}}}^2 + (1-b^2)\lambda_{\text{new}}^+ + \epsilon) + & k = 1 \\ \zeta_{j,*}^+\kappa_s^+\left(\frac{b^2}{\alpha(1-b^2)}\sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma + 4\epsilon\right) + \\ \zeta_{j,*}^+\left(\frac{b^2}{\alpha(1-b^2)}\sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma + 4\epsilon\right)\Big]\phi^+ \\ \\ \Big[(\zeta_{j,*}^+)^2(b^2 r\gamma^2 + (1-b^2)\lambda^+ + \epsilon) \\ +\zeta_{j,\text{new},k-1}^+(b^2 {c_{\text{new}}\gamma_{\text{new}}}^2 + (1-b^2)\lambda_{\text{new}}^+ + \epsilon) & k \geq 2 \\ \zeta_{j,*}^+\zeta_{j,\text{new},k-1}^+\left(\frac{b^2}{\alpha(1-b^2)}\sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma + 4\epsilon\right) \\ +\zeta_{j,*}^+\left(\frac{b^2}{\alpha(1-b^2)}\sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma + 4\epsilon\right)\Big]\left(\sqrt{\rho^2 h^+}\phi^+\right) \end{cases}$$

and

$$b_6 := (\zeta_{j,*}^+)^2(b^2 r\gamma^2 + (1-b^2)\lambda^+ + \epsilon) + \zeta_{j,*}^+\left(\frac{b^2}{\alpha(1-b^2)}\sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma + 4\epsilon\right).$$

Lemma 3.5.22. *For $k = 1, \dots, K$,*

$$\mathbb{P}\left(\|\boldsymbol{\mathcal{H}}_{\hat{u}_j+k}\|_2 \le b_{\boldsymbol{\mathcal{H}},k} + \epsilon \;\middle|\; X_{\hat{u}_j+k-1}\right) \ge 1 - p_{\boldsymbol{\mathcal{H}}} \tag{3.14}$$

for all $X_{\hat{u}_j+k-1} \in \Gamma_{j,k-1}^{\hat{u}_j}$ with $\hat{u}_j = u_j$ or $\hat{u}_j = u_j + 1$

The same bound (with $k = 1$) holds for $\|\boldsymbol{\mathcal{H}}_{u_j+1}\|_2$ when we condition on $X_{u_j} \in \Gamma_{j-1,\text{end}}$.

The above lemmas are proved in the next section (Section 3.6). The proofs use Fact 3.5.13.

3.5.4 Proofs of Lemmas 3.5.11, 3.5.12, and 3.5.14

Proof of Lemma 3.5.11. Recall that $\Gamma_{j,\text{end}} := \left(\Gamma_{j,K}^{u_j} \cap \text{NODETS}_j^{u_j}\right) \cup \left(\Gamma_{j,K}^{u_j+1} \cap \text{NODETS}_j^{u_j+1}\right)$.

1. By the definition of $\Gamma_{j,K}^{\hat{u}_j}$ for $\hat{u} = u_j$ or $\hat{u} = u_j + 1$, $\zeta_{j,*} \le \zeta_{j,*}^+ = (r_0 + (j-1)c_{\text{new}})\zeta$ and $\zeta_{j,K} \le \zeta_{j,\text{new},K}^+$. Lemma 3.5.9 and the choice of K imply that $\zeta_{j,\text{new},K}^+ \le c_{\text{new}}\zeta$. Thus, $\zeta_{j+1,*} \le \zeta_{j+1,*}^+ = (r_0 + jc_{\text{new}})\zeta$. This holds for $\hat{u}_j = u_j$ or $\hat{u}_j = u_j + 1$.

2. $\mathbb{P}(\text{NODETS}_j^{\hat{u}_j} \mid \Gamma_{j,K}^{\hat{u}_j}) = \mathbb{P}\Big(\lambda_{\max}(\boldsymbol{\mathcal{M}}_u) < \text{thresh for all } u \in [u_j + K + 1, u_{j+1} - 1] \mid \Gamma_{j,K}^{\hat{u}_j}\Big)$ for $\hat{u}_j = u_j$ or $\hat{u}_j = u_j + 1$.

As shown in 1), $\Gamma_{j,K}^{\hat{u}_j}$ implies that $\text{dif}(\hat{\boldsymbol{P}}_{(j+1),*}, \boldsymbol{P}_{(j+1),*}) \le \zeta_{j+1,*}^+ = (r_0 + jc_{\text{new}})\zeta$. Notice that for $u \in [\hat{u}_j + K + 1, u_{j+1} - 1]$, $\hat{\boldsymbol{P}}_{u\alpha-1,*} = \hat{\boldsymbol{P}}_{(j+1),*}$, $\boldsymbol{P}_{(j+1),*} = \boldsymbol{P}_{(j)}$, and $\boldsymbol{\ell}_t = \boldsymbol{P}_{(j)}\boldsymbol{a}_t$. Therefore,

$$\begin{aligned}
\lambda_{\max}(\boldsymbol{\mathcal{M}}_u) &= \lambda_{\max}\left(\frac{1}{\alpha}\sum_{t\in\mathcal{J}_u}(\boldsymbol{I} - \hat{\boldsymbol{P}}_{u\alpha-1,*}\hat{\boldsymbol{P}}_{u\alpha-1,*}{}')\hat{\boldsymbol{\ell}}_t\hat{\boldsymbol{\ell}}_t{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{u\alpha-1,*}\hat{\boldsymbol{P}}_{u\alpha-1,*}{}')\right) \\
&= \lambda_{\max}\Bigg(\frac{1}{\alpha}\sum_{t\in\mathcal{J}_u}(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j+1),*}\hat{\boldsymbol{P}}_{(j+1),*}{}')(\boldsymbol{P}_{(j)}\boldsymbol{a}_t - \boldsymbol{e}_t) \\
&\qquad\qquad (\boldsymbol{P}_{(j)}\boldsymbol{a}_t - \boldsymbol{e}_t)'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j+1),*}\hat{\boldsymbol{P}}_{(j+1),*}{}')\Bigg) \\
&\le (\zeta_{j+1,*}^+)^2 r\gamma^2 + 2\phi^+(\zeta_{j+1,*}^+)^2 r\gamma^2 + (\phi^+)^2(\zeta_{j+1,*}^+)^2 r\gamma^2 \\
&\le 4(\phi^+)^2\zeta\lambda^- \\
&\le \frac{\lambda^-}{2}.
\end{aligned}$$

The bounds on $\boldsymbol{e}_t$ come from Lemma 3.5.10, and the penultimate line uses the bounds assumed on ζ in Theorem 3.2.15.

⊠

Fact 3.5.23. *Using the bounds on* ζ, $b \leq 0.1$, $\lambda^+_{\text{new}} \leq \sqrt{2}\lambda^-$, $c_{\text{new}}\eta_{\text{new}}\lambda^+_{\text{new}} \leq 50\lambda^-$ *from the Theorem and Fact 3.5.8 (which gives* $\frac{1}{\alpha} \leq (c_{\text{new}}\zeta)^2$*), we get:*

$$b_{\boldsymbol{A}} \geq 0.94\lambda^-_{\text{new}} \geq 0.94\lambda^-$$

$$b_{\boldsymbol{A},\perp} \leq 0.011\lambda^-$$

$$b_{\boldsymbol{\mathcal{H}},k} \leq 0.24\lambda^-.$$

Thus, $b_{\boldsymbol{A}} - b_{\boldsymbol{\mathcal{H}},k} > 0.5\lambda^- = \text{thresh}$ *and* $b_{\boldsymbol{A},\perp} + b_{\boldsymbol{\mathcal{H}},k} < \text{thresh}$.

Proof of Lemma 3.5.12. We will prove that $\mathbb{P}\left(\text{DET}^{u_j+1} \mid X_{u_j}\right) > p_{\text{det},1}$ for all $X_{u_j} \in \Gamma_{j-1,\text{end}}$. In particular, this will imply that $\mathbb{P}(\text{DET}^{u_j+1} \mid X_{u_j}) > p_{\text{det},1}$ for all $X_{u_j} \in \Gamma_{j-1,\text{end}} \cap \overline{\text{DET}^{u_j}}$ and so we can conclude that $\mathbb{P}(\text{DET}^{u_j+1} \mid \Gamma_{j-1,\text{end}}, \overline{\text{DET}^{u_j}}) > p_{\text{det},1}$.

Observe that

$$\mathbb{P}\left(\text{DET}^{u_j+1} \mid X_{u_j}\right) = \mathbb{P}\left(\lambda_{\max}(\boldsymbol{\mathcal{M}}_{u_j+1}) > \text{thresh} \mid X_{u_j}\right)$$

By Weyl's Theorem

$$\begin{aligned}
\lambda_{\max}(\boldsymbol{\mathcal{M}}_{u_j+1}) &\geq \lambda_{\max}(\boldsymbol{\mathcal{A}}_{u_j+1}) + \lambda_{\min}(\boldsymbol{\mathcal{H}}_{u_j+1}) \\
&\geq \lambda_{\max}(\boldsymbol{A}_{u_j+1}) - \|\boldsymbol{\mathcal{H}}_{u_j+1}\|_2 \\
&\geq \lambda_{\min}(\boldsymbol{A}_{u_j+1}) - \|\boldsymbol{\mathcal{H}}_{u_j+1}\|_2 \\
&\geq b_{\boldsymbol{A}} - b_{\boldsymbol{\mathcal{H}},1} \\
&\geq \frac{\lambda^-}{2}
\end{aligned}$$

When $X_{u_j} \in \Gamma_{j-1,\text{end}}$, Lemmas 3.5.20 and 3.5.22 give high probability bounds on $\lambda_{\min}(\boldsymbol{A}_{u_j+1})$ and $\|\boldsymbol{\mathcal{H}}_{u_j+1}\|_2$ respectively. So the above inequality holds with probability greater than or equal to $1 - p_{\boldsymbol{A}} - p_{\boldsymbol{\mathcal{H}}} = p_{\text{det},1}$. ⊠

Proof of Lemma 3.5.14. To prove this Lemma we need to show two things. First, conditioned on $\Gamma^{\hat{u}_j}_{j,k-1}$, the k^{th} estimate of the number of new directions is correct. That is: $\hat{c}_{j,\text{new},k} = c_{j,\text{new}}$. Second, we must show $\zeta_{j,\text{new},k} \leq \zeta^+_{j,\text{new},k}$, again conditioned on $\Gamma^{\hat{u}_j}_{j,k-1}$.

Notice that $\hat{c}_{j,\text{new},k} = \text{rank}(\hat{\boldsymbol{P}}_{(j),\text{new},k})$. To show that $\text{rank}(\hat{\boldsymbol{P}}_{(j),\text{new},k}) = c_{j,\text{new}}$, we need to show that for $u = \hat{u}_j + k$, $k = 1, \ldots, K$, $\lambda_{c_{j,\text{new}}}(\boldsymbol{\mathcal{M}}_u) > \text{thresh}$ and $\lambda_{c_{j,\text{new}}+1}(\boldsymbol{\mathcal{M}}_u) < \text{thresh}$. To do this we proceed similarly to above. Observe that, $\boldsymbol{\mathcal{M}}_u = \boldsymbol{\mathcal{A}}_u + \boldsymbol{\mathcal{H}}_u$. By Weyl's Theorem

$$\begin{aligned}
\lambda_{c_{j,\text{new}}}(\boldsymbol{\mathcal{M}}_u) &\geq \lambda_{c_{j,\text{new}}}(\boldsymbol{\mathcal{A}}_u) + \lambda_{\min}(\boldsymbol{\mathcal{H}}_u) \\
&\geq \lambda_{c_{j,\text{new}}}(\boldsymbol{\mathcal{A}}_u) - \|\boldsymbol{\mathcal{H}}_u\|_2 \\
&= \lambda_{\min}(\boldsymbol{A}_u) - \|\boldsymbol{\mathcal{H}}_u\|_2.
\end{aligned}$$

The equality is because $\boldsymbol{A}_u$ is of size $c_{j,\text{new}} \times c_{j,\text{new}}$ and $\lambda_{\min}(\boldsymbol{A}_u) > \lambda_{\max}(\boldsymbol{A}_{u,\perp})$. Similarly,

$$\begin{aligned}
\lambda_{c_{j,\text{new}}+1}(\boldsymbol{\mathcal{M}}_u) &\leq \lambda_{c_{j,\text{new}}+1}(\boldsymbol{\mathcal{A}}_u) + \lambda_{\max}(\boldsymbol{\mathcal{H}}_u) \\
&\leq \lambda_{c_{j,\text{new}}+1}(\boldsymbol{\mathcal{A}}_u) + \|\boldsymbol{\mathcal{H}}_u\|_2 \\
&= \lambda_{\max}(\boldsymbol{A}_{u,\perp}) + \|\boldsymbol{\mathcal{H}}_u\|_2.
\end{aligned}$$

Using Lemmas 3.5.20, 3.5.21, and 3.5.22 and Fact 3.5.23, we can conclude that with probability greater than $1 - p_{\text{ppca}}$, $\lambda_{c_{j,\text{new}}}(\boldsymbol{\mathcal{M}}_u) > b_{\boldsymbol{A}} - b_{\boldsymbol{\mathcal{H}},k} \geq \lambda^-/2 = \text{thresh}$ and $\lambda_{c_{j,\text{new}}+1}(\boldsymbol{\mathcal{M}}_u) < b_{\boldsymbol{A},\perp} + b_{\boldsymbol{\mathcal{H}},k} \leq \lambda^-/2 = \text{thresh}$. Therefore $\text{rank}(\hat{\boldsymbol{P}}_{(j),\text{new},k}) = c_{j,\text{new}}$ with probability greater than $1 - p_{\text{ppca}}$.

To show that $\zeta_{j,\text{new},k} \leq \zeta^+_{k,\text{new}}$, we also use Lemmas 3.5.20, 3.5.21, and 3.5.22. Using $\text{rank}(\hat{\boldsymbol{P}}_{(j),\text{new},k}) = c_{j,\text{new}}$ and applying Lemma 3.5.17 with these bounds gives the desired result.

⊠

3.6 Proofs of Lemmas 3.5.20, 3.5.21, and 3.5.22

3.6.1 Key Lemmas Needed for the Proofs

Recall from the notation section that for a sequence of random variables $\boldsymbol{Z}_t$, we use the notation $\mathbb{E}_{t-1}[\boldsymbol{Z}_t]$ to mean the expectation of $\boldsymbol{Z}_t$ conditioned on all of the previous $\boldsymbol{Z}_t$'s. That is:

$$\mathbb{E}_{t-1}[\boldsymbol{Z}_t] := \mathbb{E}[\boldsymbol{Z}_t | \boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_{t-1}]$$

and

$$\mathbb{E}_{t-1}[\boldsymbol{Z}_t \big| X] := \mathbb{E}[\boldsymbol{Z}_t \big| X, \boldsymbol{Z}_1, \ldots, \boldsymbol{Z}_{t-1}].$$

Lemma 3.6.1. *For $j = 1, \dots, J$ and $k = 1, \dots, K$, for all $X_{\hat{u}_j+k-1} \in \Gamma_{j,k-1}^{\hat{u}_j}$*

1. $\mathbf{0} \preceq \frac{1}{\alpha} \sum_{t \in \mathcal{J}_{\hat{u}_j+k}} \mathbb{E}_{t-1}\left[\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,*}{}' \;\middle|\; X_{\hat{u}_j+k-1}\right] = b^2 \boldsymbol{a}_{t-1,*}\boldsymbol{a}_{t-1,*}{}' + (\boldsymbol{\Lambda}_{\nu,t})_* \preceq \left(b^2 r\gamma^2 + (1 - b^2)\lambda^+\right)\boldsymbol{I}$
2. $(1-b^2)\lambda_{\text{new}}^- \boldsymbol{I} \preceq \frac{1}{\alpha} \sum_{t \in \mathcal{J}_{\hat{u}_j+k}} \mathbb{E}_{t-1}\left[\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,\text{new}}{}' \;\middle|\; X_{\hat{u}_j+k-1}\right] = b^2 \boldsymbol{a}_{t-1,\text{new}}\boldsymbol{a}_{t-1,\text{new}}{}' + (\boldsymbol{\Lambda}_{\nu,t})_{\text{new}} \preceq \left(b^2 c_{\text{new}}\gamma_{\text{new}}{}^2 + (1 - b^2)\lambda_{\text{new}}^+\right)\boldsymbol{I}$
3. $\frac{1}{\alpha} \sum_{t \in \mathcal{J}_{\hat{u}_j+k}} \mathbb{E}_{t-1}\left[\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,\text{new}}{}' \;\middle|\; X_{\hat{u}_j+k-1}\right] = b^2 \boldsymbol{a}_{t-1,*}\boldsymbol{a}_{t-1,\text{new}}{}'$

with $\hat{u}_j = u_j$ or $\hat{u}_j = u_j + 1$. The only reason we need the assumption $X_{\hat{u}_j+k-1} \in \Gamma_{j,k-1}^{\hat{u}_j}$ is to apply Fact 3.5.13 which allows us to use the "$\boldsymbol{a}_{t,\text{new}}$ small" bounds.

The same bounds also hold for summation over $t \in \mathcal{J}_{u_j+1}$ when we condition on $X_{u_j} \in \Gamma_{j-1,\text{end}}$.

The proof is given in Section 3.7. We implicitly use Fact 3.5.13 in the proof.

Using $\|\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,*}{}'\|_2 \le r\gamma^2$, $\|\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,\text{new}}{}'\|_2 \le c\gamma_{\text{new}}{}^2$ (this holds because of Fact 3.5.13) and applying the matrix Azuma inequality (Lemma 3.A.9 in the appendix) to the first two claims above gives the following lemma.

Lemma 3.6.2. *For $j = 1, \dots, J$ and $k = 1, \dots, K$,*

1. $\mathbb{P}\left(\lambda_{\max}\left(\frac{1}{\alpha}\sum_{t \in \mathcal{J}_{\hat{u}_j+k}} \boldsymbol{a}_{t,*}\boldsymbol{a}_{t,*}{}'\right) \le b^2 r\gamma^2 + (1 - b^2)\lambda^+ + \epsilon \;\middle|\; X_{\hat{u}_j+k-1}\right) \ge 1 - r\digamma(\alpha, \epsilon, r\gamma^2)$
2. $\mathbb{P}\left(\lambda_{\min}\left(\frac{1}{\alpha}\sum_{t \in \mathcal{J}_{\hat{u}_j+k}} \boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,\text{new}}{}'\right) \ge (1 - b^2)\lambda_{\text{new}}^- - \epsilon \;\middle|\; X_{\hat{u}_j+k-1}\right) \ge 1 - r\digamma(\alpha, \epsilon, c_{\text{new}}\gamma_{\text{new}}{}^2)$
3. $\mathbb{P}\left(\lambda_{\max}\left(\frac{1}{\alpha}\sum_{t \in \mathcal{J}_{\hat{u}_j+k}} \boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,\text{new}}{}'\right) \le b^2 c_{\text{new}}\gamma_{\text{new}}{}^2 + (1 - b^2)\lambda_{\text{new}}^+ + \epsilon \;\middle|\; X_{\hat{u}_j+k-1}\right) \ge 1 - r\digamma(\alpha, \epsilon, c_{\text{new}}\gamma_{\text{new}}{}^2)$

for all $X_{\hat{u}_j+k-1} \in \Gamma_{j,0}^{\hat{u}_j}$ with $\hat{u}_j = u_j$ or $\hat{u}_j = u_j + 1$.

The same bounds also hold for summation over $t \in \mathcal{J}_{u_j+1}$ when we condition on $X_{u_j} \in \Gamma_{j-1,\text{end}}$.

Lemma 3.6.3.

$$\mathbb{P}\left(\left\|\frac{1}{\alpha}\sum_{t\in\mathcal{J}_{\hat{u}_j+k}}\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,*}{}'\right\|_2 \le \frac{b^2}{\alpha(1-b^2)}\sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma + 4\epsilon \;\middle|\; X_{\hat{u}_j+k-1}\right) \ge$$

$$1 - 3(r+c_{\text{new}})F(\alpha,\epsilon,2\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}) - (r+c_{\text{new}})F(\alpha,\epsilon,4\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}).$$

for all $X_{\hat{u}_j+k-1}\in\Gamma_{j,0}^{\hat{u}_j}$ *with* $\hat{u}_j = u_j$ *or* $\hat{u}_j = u_j+1$.

The same bounds also hold for summation over $t\in\mathcal{J}_{u_j+1}$ *when we condition on* $X_{u_j}\in\Gamma_{j-1,\text{end}}$.

The proof is also in Section 3.7. We use Fact 3.5.13 in the proof.

Remark 3.6.4. *Whenever Lemma 3.6.2 or 3.6.3 is applied, we set* $\epsilon = 0.01c_{\text{new}}\zeta\lambda^-$.

Lemma 3.6.2 follows directly from Lemma 3.6.1 and the matrix Azuma inequality. Lemma 3.6.3 needs more work to get the $\frac{1}{\alpha}$ factor. This is needed because γ can be large, and the cross term $\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,*}{}'$ will appear later without an appropriately small factor multiplying it.

Remark 3.6.5. *It is possible to also bound* $\lambda_{\max}(\frac{1}{\alpha}\sum_t \boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,\text{new}}{}')$ *and* $\lambda_{\max}(\frac{1}{\alpha}\sum_t \boldsymbol{a}_{t,*}\boldsymbol{a}_{t,*}{}')$ *using the same approach used in Lemma 3.6.3. This would give a* $\frac{1}{\alpha}$ *multiplying the* $b^2c_{\text{new}}\gamma_{\text{new}}{}^2$ *terms.*

Lemma 3.6.6. *Assume that the assumptions of Theorem 3.2.15 hold. Conditioned on* $X_{\hat{u}_j+k-1}\in\Gamma_{j,k-1}^{\hat{u}_j}$ *for* $\hat{u}_j = u_j$ *or* $\hat{u}_j = u_j+1$,

$$\|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{D}_{j,\text{new}}\|_2 \le \kappa_s^+ := .0215 \tag{3.15}$$

for all $\mathcal{T}$ *such that* $|\mathcal{T}|\le s$.

Proof. Recall that $\boldsymbol{D}_{j,\text{new}} = (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{P}_{(j),\text{new}}$. Then $\|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{D}_{j,\text{new}}\|_2 = \|\boldsymbol{I}_{\mathcal{T}}{}'(\boldsymbol{I}-\hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{P}_{(j),\text{new}}\|_2 \le \|\boldsymbol{I}_{\mathcal{T}}{}'\boldsymbol{P}_{(j),\text{new}}\|_2 + \|\hat{\boldsymbol{P}}_{(j),*}{}'\boldsymbol{P}_{(j),\text{new}}\|_2 \le \kappa_s(\boldsymbol{P}_{(j),\text{new}}) + \|\hat{\boldsymbol{P}}_{(j),*}{}'(\boldsymbol{I}-\boldsymbol{P}_{(j),*}\boldsymbol{P}_{(j),*}{}')\boldsymbol{P}_{(j),\text{new}}\|_2 \le \kappa_s(\boldsymbol{P}_{(j),\text{new}}) + \zeta_{j,*}$. The event $X_{\hat{u}_j+k-1}\in\Gamma_{j,k-1}^{\hat{u}_j}$ implies that $\zeta_{j,*}\le\zeta_{j,*}^+\le 0.0015$. Thus, the lemma follows. ⊠

3.6.2 Proofs of Lemmas 3.5.20, 3.5.21, 3.5.22

Definition 3.6.7. *Define the following*

1. $\hat{\boldsymbol{P}}_{(j),\text{new},0} = [.]$ *(empty matrix)*
2. $\boldsymbol{D}_{j,\text{new},k} := (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}' - \hat{\boldsymbol{P}}_{(j),\text{new},k}\hat{\boldsymbol{P}}_{(j),\text{new},k}{}')\boldsymbol{P}_{(j),\text{new}}$. *Recall that* $\boldsymbol{D}_{j,\text{new}} := \boldsymbol{D}_{j,\text{new},0}$.
3. $\boldsymbol{D}_{j,*,k} := (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}' - \hat{\boldsymbol{P}}_{(j),\text{new},k}\hat{\boldsymbol{P}}_{(j),\text{new},k}{}')\boldsymbol{P}_{(j),*}$ *and* $\boldsymbol{D}_{j,*} := \boldsymbol{D}_{j,*,0}$
4. *Recall that* $\zeta_{j,0} = \|\boldsymbol{D}_{j,\text{new}}\|_2$, $\zeta_{j,\text{new},k} = \|\boldsymbol{D}_{j,\text{new},k}\|_2$, $\zeta_{j,*} = \|\boldsymbol{D}_{j,*}\|_2$. *Also, clearly,* $\|\boldsymbol{D}_{j,*,k}\|_2 \leq \zeta_{j,*}$.

Remark 3.6.8. *In the rest of this section, for ease of notation, we do the following.*

- *We remove the subscript* j *from* $\boldsymbol{D}_{j,\text{new},k}$ *etc.,* $\boldsymbol{E}_{j,\text{new}}$ *etc. and* $\zeta_{j,\text{new},k}$ *etc. (from everything in Definitions 3.5.3 and 3.5.16).*
- *Similarly we also let* $X_k := X_{\hat{u}_j+k}$ *and* $\Gamma_k := \Gamma_{j,k}^{\hat{u}_j}$. *Thus, if we say* $\mathbb{P}\left(\text{event}|X_{k-1} \in \Gamma_{k-1}\right) \geq p_0$ *we mean that* $\mathbb{P}\left(\text{event}|X_{u_j+k-1} \in \Gamma_{j,k-1}^{u_j}\right) \geq p_0$ *and* $\mathbb{P}\left(\text{event}|X_{u_j+1+k-1} \in \Gamma_{j,k-1}^{u_j+1}\right) \geq p_0$.
- *Finally,* $\sum_t$ *refers to* $\sum_{t\in\mathcal{J}_u}$ *for* $u = \hat{u}_j + k$, $k = 1, 2, \ldots, K$, $j = 1, 2, \ldots, J$.

Also, note the following.

- *Recall that* $\mathcal{T}_t$ *is included in the definition of* X_{k-1}, *so conditioned on* X_{k-1}, *the* $\mathcal{T}_t$ *are deterministic.*
- *The proof for the bound on* $\boldsymbol{A}_u$ *for* $u = u_j + 1$ *is the same as that for* $u = \hat{u}_j + 1$ *since in both cases* $\hat{\boldsymbol{P}}_{t,*} = \hat{\boldsymbol{P}}_{(j),*} = \hat{\boldsymbol{P}}_{(j-1)}$ *and* $\hat{\boldsymbol{P}}_{t,\text{new}} = [.]$ *for all* $t \in \mathcal{J}_u$. *The same is true for the bounds on* $\boldsymbol{A}_{u_j+1,\perp}$ *and* $\boldsymbol{\mathcal{H}}_{u_j+1}$.

Lemma 3.6.9. *When* $X_{k-1} \in \Gamma_{k-1}$,

1. $\|\boldsymbol{D}_{*,k-1}\|_2 \leq \zeta_{j,*}^+$ *for* $k = 1, \ldots, K$.
2. $\|\boldsymbol{D}_{\text{new},k-1}\|_2 \leq \zeta_{k-1,\text{new}}^+$ *for* $k = 1, \ldots, K$ *(by definition of* Γ_{k-1}*)*

3. $\|[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\|_2 \leq \phi^+$ *(from Lemma 3.5.10)*

4. $\lambda_{\min}(\boldsymbol{R}_{\text{new}}\boldsymbol{R}_{\text{new}}{}') \geq 1 - (\zeta_{j,*}^+)^2$

Proof of Lemma 3.5.20. We obtain the bounds on $\boldsymbol{A}_u$ for $u = \hat{u}_j + k$ for $k = 1, 2, \ldots, K$ and $\hat{u}_j = u_j$ or $u_j + 1$. For $u = \hat{u}_j + k$, recall that $\boldsymbol{A}_u := \frac{1}{\alpha}\sum_{t\in\mathcal{J}_u} \boldsymbol{E}_{j,\text{new}}{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{\ell}_t\boldsymbol{\ell}_t{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{E}_{j,\text{new}}$.

Notice that $\boldsymbol{E}_{\text{new}}{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{\ell}_t = \boldsymbol{R}_{\text{new}}\boldsymbol{a}_{t,\text{new}} + \boldsymbol{E}_{\text{new}}{}'\boldsymbol{D}_*\boldsymbol{a}_{t,*}$. Let $\boldsymbol{Z}_t = \boldsymbol{R}_{\text{new}}\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,\text{new}}{}'\boldsymbol{R}_{\text{new}}{}'$, and let $\boldsymbol{Y}_t = \boldsymbol{R}_{\text{new}}\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,*}{}'\boldsymbol{D}_*{}'\boldsymbol{E}_{\text{new}} + \boldsymbol{E}_{\text{new}}{}'\boldsymbol{D}_*\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,\text{new}}{}'\boldsymbol{R}_{\text{new}}{}'$, then

$$\boldsymbol{A}_u \succeq \frac{1}{\alpha}\sum_t \boldsymbol{Z}_t + \frac{1}{\alpha}\sum_t \boldsymbol{Y}_t \tag{3.16}$$

Consider $\sum_t \boldsymbol{Z}_t = \sum_t \boldsymbol{R}_{\text{new}}\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,\text{new}}{}'\boldsymbol{R}_{\text{new}}{}'$. With probability 1, $\|\boldsymbol{Z}_t\|_2 \leq c_{\text{new}}{\gamma_{\text{new}}}^2$. Using a theorem of Ostrowoski [27, Theorem 4.5.9], $\lambda_{\min}(\boldsymbol{Z}_t) = \lambda_{\min}(\boldsymbol{R}_{\text{new}}\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,\text{new}}{}'\boldsymbol{R}_{\text{new}}{}') \geq \lambda_{\min}(\boldsymbol{R}_{\text{new}}\boldsymbol{R}_{\text{new}}{}')\lambda_{\min}(\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,\text{new}}{}')$.

Conditioned on X_{k-1}, the matrix $\boldsymbol{R}_{\text{new}}$ is a constant. Using Lemma 3.6.2,

$$\mathbb{P}\left(\lambda_{\min}\left(\frac{1}{\alpha}\sum_t \boldsymbol{Z}_t\right) \geq (1 - (\zeta_{j,*}^+)^2)\left[(1-b^2)\lambda_{\text{new}}^- - \epsilon\right] \;\middle|\; X_{k-1}\right) \geq 1 - c_{\text{new}}F(\alpha, \epsilon, c_{\text{new}}{\gamma_{\text{new}}}^2). \tag{3.17}$$

for all $X_{k-1} \in \Gamma_{k-1}$.

Consider $\boldsymbol{Y}_t = \boldsymbol{R}_{\text{new}}\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,*}{}'\boldsymbol{D}_*{}'\boldsymbol{E}_{\text{new}} + \boldsymbol{E}_{\text{new}}{}'\boldsymbol{D}_*\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,\text{new}}{}'\boldsymbol{R}_{\text{new}}{}'$. By Lemma 3.6.3

$$\mathbb{P}\left(\lambda_{\min}\left(\frac{1}{\alpha}\sum_t \boldsymbol{Y}_t\right) \geq -2\zeta_{j,*}^+\left(\frac{b^2}{\alpha(1-b^2)}\sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma + 4\epsilon\right) \;\middle|\; X_{k-1}\right) \geq$$

$$1 - 3(r + c_{\text{new}})F(\alpha, \epsilon, 2\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}) - (r + c_{\text{new}})F(\alpha, \epsilon, 4\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}) \tag{3.18}$$

for all $X_{k-1} \in \Gamma_{k-1}$. Combining (3.17) and (3.18) the lemma follows. ⊠

Proof of Lemma 3.5.21. Remark 3.6.8 applies.

We obtain the bounds on $\boldsymbol{A}_{u,\perp}$ for $u = \hat{u}_j + k$ for $k = 1, 2, \ldots K$ and $\hat{u}_j = u_j$ or $u_j + 1$. For $u = \hat{u}_j + k$, recall that $\boldsymbol{A}_{u,\perp} := \frac{1}{\alpha}\sum_t \boldsymbol{E}_{\text{new},\perp}{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{\ell}_t\boldsymbol{\ell}_t{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{E}_{\text{new},\perp}$. By their definitions, $\boldsymbol{E}_{\text{new},\perp}{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{\ell}_t = \boldsymbol{E}_{\text{new},\perp}{}'\boldsymbol{D}_*\boldsymbol{a}_{t,*}$. Thus, $\boldsymbol{A}_{u,\perp} = \frac{1}{\alpha}\sum_t \boldsymbol{Z}_t$ with

$\boldsymbol{Z}_t = \boldsymbol{E}_{\text{new},\perp}{}'\boldsymbol{D}_*\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,*}{}'\boldsymbol{D}_*{}'\boldsymbol{E}_{\text{new},\perp}$. Conditioned on $X_{k-1} \in \Gamma_{k-1}$, $\|\boldsymbol{Z}_t\|_2 \leq (\zeta_{j,*}^+)^2 r\gamma^2$. Using Lemma 3.6.2, we get that

$$\mathbb{P}\left(\lambda_{\max}\left(\frac{1}{\alpha}\sum_t \boldsymbol{A}_{u,\perp}\right) \leq (\zeta_{j,*}^+)^2(b^2 r\gamma^2 + (1-b^2)\lambda^+ + \epsilon) \;\middle|\; X_{k-1}\right) \geq 1 - r\digamma\left(\alpha, \epsilon, (\zeta_{j,*}^+)^2 r\gamma^2\right) \tag{3.19}$$

for all $X_{k-1} \in \Gamma_{k-1}$. ⊠

Proof of Lemma 3.5.22. Remark 3.6.8 applies.

Consider the $\boldsymbol{\mathcal{H}}_u$ term. For ease of notation, define

$$\tilde{\boldsymbol{\ell}}_t = (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{\ell}_t.$$

Using the expression for $\boldsymbol{\mathcal{H}_u}$ given in Definition 3.5.16, and noting that for a basis matrix $\boldsymbol{E}$, $\boldsymbol{E}\boldsymbol{E}' + \boldsymbol{E}_\perp\boldsymbol{E}_\perp{}' = \boldsymbol{I}$ we get that

$$\begin{aligned}\boldsymbol{\mathcal{H}}_u = \frac{1}{\alpha}\sum_{t\in\mathcal{J}_u}\Big(&(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{e}_t\boldsymbol{e}_t{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}') \\ &- \left(\tilde{\boldsymbol{\ell}}_t\boldsymbol{e}_t{}'(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}') + (\boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}')\boldsymbol{e}_t\tilde{\boldsymbol{\ell}}_t{}'\right) + \left(\boldsymbol{F}_t + \boldsymbol{F}_t{}'\right)\Big)\end{aligned}$$

where

$$\boldsymbol{F}_t = \boldsymbol{E}_{\text{new},\perp}\boldsymbol{E}_{\text{new},\perp}{}'\tilde{\boldsymbol{\ell}}_t\tilde{\boldsymbol{\ell}}_t{}'\boldsymbol{E}_{\text{new}}\boldsymbol{E}_{\text{new}}{}'.$$

Thus,

$$\|\boldsymbol{\mathcal{H}}_u\|_2 \leq \left\|\frac{1}{\alpha}\sum_t \boldsymbol{e}_t\boldsymbol{e}_t{}'\right\|_2 + 2\left\|\frac{1}{\alpha}\sum_t \tilde{\boldsymbol{\ell}}_t\boldsymbol{e}_t{}'\right\|_2 + 2\left\|\frac{1}{\alpha}\sum_t \boldsymbol{F}_t\right\|_2 \tag{3.20}$$

Next we obtain high probability bounds on each of the three terms on the right hand side of (3.20).

Consider $\|\frac{1}{\alpha}\sum_t \boldsymbol{e}_t\boldsymbol{e}_t{}'\|_2$. Using Lemma 3.5.10, $\boldsymbol{e}_t$ satisfies (3.9) with probability one for all $X_{k-1} \in \Gamma_{k-1}$. Expanding this expression gives

$$\boldsymbol{e}_t\boldsymbol{e}_t{}' = \left(\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{\Phi}_t\boldsymbol{\ell}_t\right)\left(\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{\Phi}_t\boldsymbol{\ell}_t\right)'$$

From Lemma 3.5.10, conditioned on $X_{k-1} \in \Gamma_{k-1}$, $\|\boldsymbol{e}_t\boldsymbol{e}_t{}'\|_2 \leq \left[\phi^+\left(\zeta_{j,*}^+\sqrt{r}\gamma + \sqrt{c_{\text{new}}}\gamma_{\text{new}}\right)\right]^2$. This is a looser but simpler bound obtaining by using $\zeta_{j,\text{new},k-1}^+ \leq 1$.

We can further decompose $\boldsymbol{e}_t\boldsymbol{e}_t{}'$ as

$$\boldsymbol{e}_t\boldsymbol{e}_t{}' = \mathbf{term}_{1,t} + \mathbf{term}_{2,t} + \mathbf{term}_{3,t} + \mathbf{term}_{3,t}{}'$$

where

$$\mathbf{term}_{1,t} = \boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{D}_{*,k-1}\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,*}{}'\boldsymbol{D}_{*,k-1}{}'\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'$$

$$\mathbf{term}_{2,t} = \boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{D}_{\text{new},k-1}\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,\text{new}}{}'\boldsymbol{D}_{\text{new},k-1}{}'\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'$$

$$\mathbf{term}_{3,t} = \boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{D}_{*,k-1}\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,\text{new}}{}'\boldsymbol{D}_{\text{new},k-1}{}'\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'$$

Then by Lemma 3.6.1,

$$\mathbb{E}_{t-1}[\mathbf{term}_{1,t}|X_{k-1}] = \boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'$$
$$\boldsymbol{D}_{*,k-1}\big(b^2\boldsymbol{a}_{t-1,*}\boldsymbol{a}_{t-1,*}{}' + (\boldsymbol{\Lambda}_{\nu,t})_*\big)\boldsymbol{D}_{*,k-1}{}'\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'$$

$$\mathbb{E}_{t-1}[\mathbf{term}_{2,t}|X_{k-1}] = \boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{D}_{\text{new},k-1}\big(b^2\boldsymbol{a}_{t-1,\text{new}}\boldsymbol{a}_{t-1,\text{new}}{}' + (\boldsymbol{\Lambda}_{\nu,t})_{\text{new}}\big)$$
$$\boldsymbol{D}_{\text{new},k-1}{}'\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'$$

$$\mathbb{E}_{t-1}[\mathbf{term}_{3,t}|X_{k-1}] = \boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'$$
$$\boldsymbol{D}_{*,k-1}b^2\boldsymbol{a}_{t-1,*}\boldsymbol{a}_{t-1,\text{new}}{}'\boldsymbol{D}_{\text{new},k-1}{}'\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'$$

The following uses Signal Model 3.3.1, Lemma 3.3.3, and Lemma 3.6.1. When $k = 1$ we use Lemma 3.6.6 which gives the bound $\|\boldsymbol{D}_{\text{new},0}{}'\boldsymbol{I}_{\mathcal{T}_t}\|_2 \leq \kappa_s^+$:

$$\left\|\frac{1}{\alpha}\sum_t \mathbb{E}_{t-1}[\mathbf{term}_{1,t}|X_{k-1}]\right\|_2 \leq \rho^2 h^+(\phi^+)^2(\zeta_{j,*}^+)^2(b^2r\gamma^2 + (1-b^2)\lambda^+)$$
$$\left\|\frac{1}{\alpha}\sum_t \mathbb{E}_{t-1}[\mathbf{term}_{2,t}|X_{k-1}]\right\|_2 \leq \rho^2 h^+(\phi^+)^2(\kappa_s^+)^2(b^2c_{\text{new}}{\gamma_{\text{new}}}^2 + (1-b^2)\lambda_{\text{new}}^+)$$
$$\left\|\frac{1}{\alpha}\sum_t \mathbb{E}_{t-1}[\mathbf{term}_{3,t}|X_{k-1}]\right\|_2 \leq \rho^2 h^+(\phi^+)^2\kappa_s^+\zeta_{j,*}^+b^2\sqrt{rc_{\text{new}}}\gamma\gamma_{\text{new}}.$$

And when $k \geq 2$,

$$\left\|\frac{1}{\alpha}\sum_t \mathbb{E}_{t-1}[\mathbf{term}_{1,t}|X_{k-1}]\right\|_2 \leq \rho^2 h^+(\phi^+)^2(\zeta_{j,*}^+)^2(b^2r\gamma^2 + (1-b^2)\lambda^+)$$
$$\left\|\frac{1}{\alpha}\sum_t \mathbb{E}_{t-1}[\mathbf{term}_{2,t}|X_{k-1}]\right\|_2 \leq \rho^2 h^+(\phi^+)^2(\zeta_{j,\text{new},k-1}^+)^2(b^2c_{\text{new}}{\gamma_{\text{new}}}^2 + (1-b^2)\lambda_{\text{new}}^+)$$
$$\left\|\frac{1}{\alpha}\sum_t \mathbb{E}_{t-1}[\mathbf{term}_{3,t}|X_{k-1}]\right\|_2 \leq \rho^2 h^+(\phi^+)^2\zeta_{j,*}^+\zeta_{j,\text{new},k-1}^+b^2\sqrt{rc_{\text{new}}}\gamma\gamma_{\text{new}}.$$

Thus by Lemma 3.A.10 (Azuma Corollary)

$$\mathbb{P}\left(\left\|\frac{1}{\alpha}\sum_t \boldsymbol{e}_t\boldsymbol{e}_t'\right\|_2 \le b_{2,k}+\epsilon \;\middle|\; X_{k-1}\right) \ge 1-n\digamma\left(\alpha,\epsilon,2\left[\phi^+\left(\zeta_{j,*}^+\sqrt{r}\gamma+\sqrt{c_{\text{new}}}\gamma_{\text{new}}\right)\right]^2\right) \tag{3.21}$$

for all $X_{k-1}\in\Gamma_{k-1}$.

Next, consider $\left\|\frac{1}{\alpha}\sum_t \tilde{\boldsymbol{\ell}}_t\boldsymbol{e}_t'\right\|_2$. Observe that when $X_{k-1}\in\Gamma_{k-1}$,

$$\begin{aligned}\tilde{\boldsymbol{\ell}}_t\boldsymbol{e}_t' &= (\boldsymbol{I}-\hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}')(\boldsymbol{P}_*\boldsymbol{a}_{t,*}+\boldsymbol{P}_{\text{new}}\boldsymbol{a}_{t,\text{new}})(\boldsymbol{P}_*\boldsymbol{a}_{t,*}+\boldsymbol{P}_{\text{new}}\boldsymbol{a}_{t,\text{new}})'\boldsymbol{\Phi}_t'\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}'\\ &= (\boldsymbol{D}_*\boldsymbol{a}_{t,*}+\boldsymbol{D}_{\text{new}}\boldsymbol{a}_{t,\text{new}})(\boldsymbol{D}_{*,k-1}\boldsymbol{a}_{t,*}+\boldsymbol{D}_{\text{new},k-1}\boldsymbol{a}_{t,\text{new}})'\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}'\end{aligned}$$

Redefine

$$\begin{aligned}\mathbf{term}_{1,t} &:= \left(\boldsymbol{D}_*\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,*}'\boldsymbol{D}_{*,k-1}'+\boldsymbol{D}_{\text{new}}\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,\text{new}}'\boldsymbol{D}_{\text{new},k-1}'\right)\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}'\\ \mathbf{term}_{2,t} &:= \left(\boldsymbol{D}_*\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,\text{new}}'\boldsymbol{D}_{\text{new},k-1}'+\boldsymbol{D}_{\text{new}}\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,*}'\boldsymbol{D}_{*,k-1}'\right)\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}'\end{aligned}$$

When $k=1$ we use Lemma 3.6.6 which gives the bound $\|\boldsymbol{D}_{\text{new}}'\boldsymbol{I}_{\mathcal{T}_t}\|_2\le\kappa_s^+$. By Lemma 3.6.2,

$$\begin{aligned}\mathbb{P}\left(\left\|\frac{1}{\alpha}\sum_{t\in\mathcal{J}_{\hat{u}_j+1}}\mathbf{term}_{1,t}\right\|_2 \le \Big[(\zeta_{j,*}^+)^2(b^2r\gamma^2+(1-b^2)\lambda^+ +\epsilon)\right.\\ \left.+\zeta_{j,\text{new},k-1}^+\kappa_s^+(b^2c_{\text{new}}{\gamma_{\text{new}}}^2+(1-b^2)\lambda_{\text{new}}^+ +\epsilon)\Big]\phi^+ \;\middle|\; X_0\right)\\ \ge 1-r\digamma\left(\alpha,\epsilon,r\gamma^2\right)-c\digamma\left(\alpha,\epsilon,c_{\text{new}}{\gamma_{\text{new}}}^2\right).\end{aligned}$$

for all $X_0\in\Gamma_0$. And by Lemma 3.6.3,

$$\begin{aligned}\mathbb{P}\left(\left\|\frac{1}{\alpha}\sum_{t\in\mathcal{J}_{\hat{u}_j+1}}\mathbf{term}_{2,t}\right\|_2 \le \left[\zeta_{j,*}^+\zeta_{j,\text{new},k-1}^+\kappa_s^+\left(\frac{b^2}{\alpha(1-b^2)}\sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma+4\epsilon\right)\right.\\ \left.+\zeta_{j,*}^+\left(\frac{b^2}{\alpha(1-b^2)}\sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma+4\epsilon\right)\right]\phi^+ \;\middle|\; X_0\right)\\ \ge 1-3(r+c_{\text{new}})\digamma\left(\alpha,\epsilon,2\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}\right)-(r+c_{\text{new}})\digamma\left(\alpha,\epsilon,4\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}\right)\end{aligned}$$

for all $X_0\in\Gamma_0$.

When $k\ge 2$ we apply Cauchy-Schwarz (Lemma 3.A.3) to $\mathbf{term}_{1,t}$. Here $\boldsymbol{X}_t=\left(\boldsymbol{D}_*\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,*}'\boldsymbol{D}_{*,k-1}'+\boldsymbol{D}_{\text{new}}\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,\text{new}}'\boldsymbol{D}_{\text{new},k-1}'\right)$ and $\boldsymbol{Y}_t=\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}'(\boldsymbol{\Phi}_t)_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}'$.

We can bound the norm of $\frac{1}{\alpha}\sum_t \boldsymbol{X}_t\boldsymbol{X}_t'$ using Lemmas 3.6.2 and 3.6.9. By Lemma 3.3.3 we get

$$\left\|\frac{1}{\alpha}\sum_t \boldsymbol{Y}_t\boldsymbol{Y}_t'\right\|_2 \leq \rho^2 h^+(\phi^+)^2$$

So when $k \geq 2$,

$$\begin{aligned}\mathbb{P}\Bigg(\left\|\frac{1}{\alpha}\sum_{t\in\mathcal{J}_{\hat{u}_j+k}} \mathbf{term}_{1,t}\right\|_2 \leq \Big[(\zeta_{j,*}^+)^2(b^2 r\gamma^2 + (1-b^2)\lambda^+ + \epsilon) + \\ \zeta_{j,\text{new},k-1}^+(b^2 {c_\text{new}}{\gamma_\text{new}}^2 + (1-b^2)\lambda_\text{new}^+ + \epsilon)\Big]\left(\sqrt{\rho^2 h^+}\phi^+\right)\Big|\ X_{k-1}\Bigg) \\ \geq 1 - rF\left(\alpha,\epsilon,r\gamma^2\right) - {c_\text{new}}F\left(\alpha,\epsilon,{c_\text{new}}{\gamma_\text{new}}^2\right).\end{aligned}$$

for all $X_{k-1} \in \Gamma_{k-1}$. Similarly, using Lemmas 3.3.3, 3.6.3, and 3.6.9,

$$\begin{aligned}\mathbb{P}\Bigg(\left\|\frac{1}{\alpha}\sum_{t\in\mathcal{J}_{\hat{u}_j+k}} \mathbf{term}_{2,t}\right\|_2 \leq \Bigg[\zeta_{j,*}^+\zeta_{j,\text{new},k-1}^+\left(\frac{b^2}{\alpha(1-b^2)}\sqrt{{c_\text{new}}r}\gamma_\text{new}\gamma + 4\epsilon\right) + \\ \zeta_{j,*}^+\left(\frac{b^2}{\alpha(1-b^2)}\sqrt{{c_\text{new}}r}\gamma_\text{new}\gamma + 4\epsilon\right)\Bigg]\left(\sqrt{\rho^2 h^+}\phi^+\right)\Big|\ X_{k-1}\Bigg) \\ \geq 1 - 3(r+{c_\text{new}})F\left(\alpha,\epsilon,2\sqrt{{c_\text{new}}r}\gamma\gamma_\text{new}\right) - (r+{c_\text{new}})F\left(\alpha,\epsilon,4\sqrt{{c_\text{new}}r}\gamma\gamma_\text{new}\right)\end{aligned}$$

for all $X_{k-1} \in \Gamma_{k-1}$.

Thus,

$$\begin{aligned}\mathbb{P}\left(\left\|\frac{1}{\alpha}\sum_t \tilde{\boldsymbol{\ell}}_t{\boldsymbol{e}_t}'\right\|_2 \leq b_{4,k}\ \middle|\ X_{k-1}\right) \geq 1 - rF\left(\alpha,\epsilon,r\gamma^2\right) - {c_\text{new}}F\left(\alpha,\epsilon,{c_\text{new}}{\gamma_\text{new}}^2\right) - \\ 3(r+{c_\text{new}})F\left(\alpha,\epsilon,2\sqrt{{c_\text{new}}r}\gamma\gamma_\text{new}\right) - (r+{c_\text{new}})F\left(\alpha,\epsilon,4\sqrt{{c_\text{new}}r}\gamma\gamma_\text{new}\right) \qquad (3.22)\end{aligned}$$

for all $X_{k-1} \in \Gamma_{k-1}$.

Finally, consider $\left\|\frac{1}{\alpha}\sum_t \boldsymbol{F}_t\right\|_2$. Notice that

$$\begin{aligned}\boldsymbol{F}_t &= \boldsymbol{E}_{\text{new},\perp}{\boldsymbol{E}_{\text{new},\perp}}'\tilde{\boldsymbol{\ell}}_t\tilde{\boldsymbol{\ell}}_t'\boldsymbol{E}_\text{new}{\boldsymbol{E}_\text{new}}' \\ &= \boldsymbol{E}_{\text{new},\perp}{\boldsymbol{E}_{\text{new},\perp}}'(\boldsymbol{D}_*\boldsymbol{a}_{t,*} + \boldsymbol{D}_\text{new}\boldsymbol{a}_{t,\text{new}})(\boldsymbol{D}_*\boldsymbol{a}_{t,*} + \boldsymbol{D}_\text{new}\boldsymbol{a}_{t,\text{new}})'\boldsymbol{E}_\text{new}{\boldsymbol{E}_\text{new}}' \\ &= \boldsymbol{E}_{\text{new},\perp}{\boldsymbol{E}_{\text{new},\perp}}'(\boldsymbol{D}_*\boldsymbol{a}_{t,*})(\boldsymbol{D}_*\boldsymbol{a}_{t,*} + \boldsymbol{D}_\text{new}\boldsymbol{a}_{t,\text{new}})'\boldsymbol{E}_\text{new}{\boldsymbol{E}_\text{new}}' \\ &= \boldsymbol{E}_{\text{new},\perp}{\boldsymbol{E}_{\text{new},\perp}}'\left(\boldsymbol{D}_*\boldsymbol{a}_{t,*}{\boldsymbol{a}_{t,*}}'{\boldsymbol{D}_*}' + \boldsymbol{D}_*\boldsymbol{a}_{t,*}{\boldsymbol{a}_{t,\text{new}}}'{\boldsymbol{D}_\text{new}}'\right)'\boldsymbol{E}_\text{new}{\boldsymbol{E}_\text{new}}'\end{aligned}$$

Then by Lemmas 3.6.2 and 3.6.3,

$$\mathbb{P}\left(\left\|\frac{1}{\alpha}\sum_t \boldsymbol{F}_t\right\|_2 \le b_6 \;\middle|\; X_{k-1}\in\Gamma_{k-1}\right) \ge 1 - c_{\text{new}}F\left(\alpha,\epsilon,r\gamma^2|X_{k-1}\right) -$$
$$3(r+c_{\text{new}})F\left(\alpha,\epsilon,2\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}\right) - (r+c_{\text{new}})F\left(\alpha,\epsilon,4\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}\right) \tag{3.23}$$

for all $X_{k-1}\in\Gamma_{k-1}$.

Using (3.20), (3.21), (3.22) and (3.23) and the union bound, for any $X_{k-1}\in\Gamma_{k-1}$, we get the result.

⊠

3.7 Proof of Lemmas 3.6.1 and 3.6.3

Proof of Lemma 3.6.2. For the first claim, begin by observing that $\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,*}{}'$ is positive semidefinite. We also have

$$\begin{aligned}
\mathbb{E}_{t-1}[\boldsymbol{a}_{t,*}\boldsymbol{a}_{t,*}{}'|X_{k-1}] &= \mathbb{E}_{t-1}[(b\boldsymbol{a}_{t-1,*}+\boldsymbol{\nu}_{t,*})(b\boldsymbol{a}_{t-1,*}+\boldsymbol{\nu}_{t,*})'|X_{k-1}] \\
&= \mathbb{E}_{t-1}[b^2\boldsymbol{a}_{t-1,*}\boldsymbol{a}_{t-1,*}{}' + b\boldsymbol{a}_{t-1,*}\boldsymbol{\nu}_{t,*}{}' + b\boldsymbol{\nu}_{t,*}\boldsymbol{a}_{t-1,*}{}' + \boldsymbol{\nu}_{t,*}\boldsymbol{\nu}_{t,*}{}'|X_{k-1}] \\
&= b^2\boldsymbol{a}_{t-1,*}\boldsymbol{a}_{t-1,*}{}' + \boldsymbol{\Lambda}_{\nu,t} \\
&\preceq \left(b^2r\gamma^2 + (1-b^2)\lambda^+\right)\boldsymbol{I}
\end{aligned}$$

The cross terms are zero because of Lemma 3.A.4. Notice that $\boldsymbol{\nu}_{t,*}$ is zero mean and independent of $\boldsymbol{\nu}_{\tau,*}$ for $\tau<t$, and $\boldsymbol{a}_{1,*},\ldots,\boldsymbol{a}_{t-1,*}$ are functions of $\{\boldsymbol{\nu}_{\tau,*}\}$ for $\tau=1,\ldots,t-1$.

Claim 2 is done in exact same manner, except that we also need the fact that $\lambda_{\min}(\boldsymbol{A}+\boldsymbol{B})\ge\lambda_{\min}(\boldsymbol{A})+\lambda_{\min}(\boldsymbol{B})$.

Claim 3 uses the same expansion and the fact that $\boldsymbol{\nu}_t$ has diagonal covariance, so $\mathbb{E}[\boldsymbol{\nu}_{t,*}\boldsymbol{\nu}_{t,\text{new}}{}'|X_{k-1}]=\boldsymbol{0}$. ⊠

Proof of Lemma 3.6.3. By Lemma 3.A.6 with $\boldsymbol{c}_\tau=\boldsymbol{a}_{\tau+(\hat{u}_j+k-1)\alpha+1,\text{new}}$, $\boldsymbol{\mu}_\tau=\boldsymbol{\nu}_{\tau+(\hat{u}_j+k-1)\alpha+1,\text{new}}$, $\tilde{\boldsymbol{c}}_\tau=\boldsymbol{a}_{\tau+(\hat{u}_j+k-1)\alpha+1,*}$ and $\tilde{\boldsymbol{\mu}}_\tau=\boldsymbol{\nu}_{\tau+(\hat{u}_j+k-1)\alpha+1,*}$ for $\tau=0,1,\ldots\alpha-1$, we get

$$\sum_{t\in\mathcal{J}_{\hat{u}_j+k}}\boldsymbol{a}_{t,\text{new}}\boldsymbol{a}_{t,*}{}' = \sum_{i\in\mathcal{J}_{\hat{u}_j+k}}[\boldsymbol{Z}_{1,i}+\boldsymbol{Z}_{2,i}+\boldsymbol{Z}_{3,i}+\boldsymbol{Z}_{4,i}]+\boldsymbol{Z}_5$$

with

$$\boldsymbol{Z}_{1,i} = \frac{(1-b^{2(\alpha-i)})}{1-b^2}\boldsymbol{\nu}_{i,\text{new}}\boldsymbol{\nu}_{i,*}{}',$$
$$\boldsymbol{Z}_{2,i} = \sum_{i_2=(\hat{u}_j+k-1)\alpha+1}^{i-1} \frac{(1-b^{2(\alpha-i)})}{1-b^2} b^{i-i_2}\boldsymbol{\nu}_{i,\text{new}}\boldsymbol{\nu}_{i_2,*}{}',$$
$$\boldsymbol{Z}_{3,i} = \sum_{i_2=(\hat{u}_j+k)\alpha+1-i}^{(\hat{u}_j+k)\alpha} \frac{(1-b^{2(\alpha-i_2)})}{1-b^2} b^{i+i_2-\alpha+1}\boldsymbol{\nu}_{\alpha-i-1,\text{new}}\boldsymbol{\nu}_{i_2,*}{}',$$
$$\boldsymbol{Z}_{4,i} = \frac{b^{i+1}(1-b^{2(\alpha-i)})}{1-b^2}(\boldsymbol{\nu}_{i,\text{new}}\boldsymbol{a}_{(\hat{u}_j+k-1)\alpha,*}{}' + \boldsymbol{a}_{(\hat{u}_j+k-1)\alpha,\text{new}}\boldsymbol{\nu}_{i,*}{}')$$
$$\boldsymbol{Z}_5 = \frac{b^2(1-b^{2\alpha})}{1-b^2}\boldsymbol{a}_{(\hat{u}_j+k-1)\alpha,\text{new}}\boldsymbol{a}_{(\hat{u}_j+k-1)\alpha,*}{}'$$

Using $\|\boldsymbol{\nu}_t\|_\infty \leq (1-b)\gamma$, $\|\boldsymbol{\nu}_{t,\text{new}}\|_\infty \leq (1-b)\gamma_{\text{new}}$, and the geometric series formula, we get the following norm bounds (recall that $b < 1$):

1. $\|\boldsymbol{Z}_{1,i}\|_2 \leq \frac{(1-b^{2\alpha})(1-b)}{1+b}\sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma \leq \sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma$
2. $\|\boldsymbol{Z}_{2,i}\|_2 \leq \sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma$
3. $\|\boldsymbol{Z}_{3,i}\|_2 \leq \sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma$
4. $\|\boldsymbol{Z}_{4,i}\|_2 \leq 2\sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma$
5. $\|\boldsymbol{Z}_5\|_2 \leq \frac{b^2}{1-b^2}\sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma$

The bounds 1-4 may be somewhat loose, but they are only needed to lower bound the probability of the good event. Bound 5) will appear in the $\zeta^+_{k,\text{new}}$ expression, so we retain the b's.

In all expectations, we need to condition on X_{k-1} for all $X_{k-1} \in \Gamma_{k-1}$ in order use Fact 3.5.13. This is necessary to ensure that the tighter bound γ_{new} applies for the given time interval. By diagonal covariance $\mathbb{E}_{i-1}[\boldsymbol{Z}_{1,i}|X_{k-1}] = \boldsymbol{0}$. The proof of $\mathbb{E}_{i-1}[\boldsymbol{Z}_{2,i}] = \mathbb{E}_{i-1}[\boldsymbol{Z}_{3,i}] = \boldsymbol{0}$ is shown below. $\mathbb{E}_{i-1}[\boldsymbol{Z}_{4,i}] = \boldsymbol{0}$ because the $\boldsymbol{\nu}_t$ are zero-mean and $\boldsymbol{a}_{(\hat{u}_j+k-1)\alpha}$ is constant conditioned on X_{k-1}. $\boldsymbol{Z}_5$ will be the non-zero term.

Ignoring the scalar coefficients,

$$\begin{aligned}
\mathbb{E}_{i-1}[\boldsymbol{Z}_{2,i}] &= \mathbb{E}_{i-1}\left[\sum_{i_2=(\hat{u}_j+k-1)\alpha+1}^{i-1} \boldsymbol{\nu}_{i,\text{new}}\boldsymbol{\nu}_{i_2,*}{}'\right] \\
&= \mathbb{E}\Bigg[\boldsymbol{\nu}_{i,\text{new}} \sum_{i_2=(\hat{u}_j+k-1)\alpha+1}^{i-1} \boldsymbol{\nu}_{i_2,*}{}' \,\Bigg|\, \\
&\qquad \sum_{i_2=(\hat{u}_j+k-1)\alpha+1}^{i-2} \boldsymbol{\nu}_{i-1,\text{new}}\boldsymbol{\nu}_{i_2,*}{}', \\
&\qquad \sum_{i_2=(\hat{u}_j+k-1)\alpha+1}^{i-3} \boldsymbol{\nu}_{i-2,\text{new}}\boldsymbol{\nu}_{i_2,*}{}', \ldots, \boldsymbol{\nu}_{(\hat{u}_j+k-1)\alpha+2,\text{new}}\boldsymbol{\nu}_{(\hat{u}_j+k-1)\alpha+1,*}{}'\Bigg] \\
&= \mathbf{0}
\end{aligned}$$

Notice that everything else in the above expression is independent of $\boldsymbol{\nu}_{i,\text{new}}$, so by Lemma 3.A.4 the expectation is zero.

Similarly (again ignore scalar coefficients for simplicity) and letting $\alpha \equiv (\hat{u}_j + k)\alpha + 1$

$$\begin{aligned}
\mathbb{E}_{i-1}[\boldsymbol{Z}_{3,i}] &= \mathbb{E}_{i-1}\left[\sum_{i_2=\alpha-i}^{\alpha-1} \boldsymbol{\nu}_{\alpha-i-1,\text{new}}\boldsymbol{\nu}_{i_2,*}{}'\right] \\
&= \mathbb{E}\Bigg[\boldsymbol{\nu}_{\alpha-i-1,\text{new}} \sum_{i_2=\alpha-i}^{\alpha-1} \boldsymbol{\nu}_{i_2,*}{}' \,\Bigg|\, \\
&\qquad \sum_{i_2=\alpha-i+1}^{\alpha-1} \boldsymbol{\nu}_{\alpha-i,\text{new}}\boldsymbol{\nu}_{i_2,*}{}', \\
&\qquad \sum_{i_2=\alpha-i+2}^{\alpha-1} \boldsymbol{\nu}_{\alpha-i+1,\text{new}}\boldsymbol{\nu}_{i_2,*}{}', \ldots, \sum_{i_2=\alpha-1}^{\alpha-1} \boldsymbol{\nu}_{\alpha-2,\text{new}}\boldsymbol{\nu}_{i_2,*}{}'\Bigg].
\end{aligned}$$

As before, all terms are independent of $\boldsymbol{\nu}_{\alpha-i-1}$ so the expectation is zero by Lemma 3.A.4.

Using the Azuma inequality,

1. $\mathbb{P}\left(\left\|\frac{1}{\alpha}\sum_{i=0}^{\alpha-1}\boldsymbol{Z}_{1,i}\right\|_2 \leq \epsilon\right) \geq 1 - (r + c_{\text{new}})F\left(\alpha, \epsilon, 2\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}\right)$
2. $\mathbb{P}\left(\left\|\frac{1}{\alpha}\sum_{i=0}^{\alpha-1}\boldsymbol{Z}_{2,i}\right\|_2 \leq \epsilon\right) \geq 1 - (r + c_{\text{new}})F\left(\alpha, \epsilon, 2\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}\right)$
3. $\mathbb{P}\left(\left\|\frac{1}{\alpha}\sum_{i=0}^{\alpha-1}\boldsymbol{Z}_{3,i}\right\|_2 \leq \epsilon\right) \geq 1 - (r + c_{\text{new}})F\left(\alpha, \epsilon, 2\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}\right)$
4. $\mathbb{P}\left(\left\|\frac{1}{\alpha}\sum_{i=0}^{\alpha-1}\boldsymbol{Z}_{4,i}\right\|_2 \leq \epsilon\right) \geq 1 - (r + c_{\text{new}})F\left(\alpha, \epsilon, 4\sqrt{c_{\text{new}}r}\gamma\gamma_{\text{new}}\right)$

5. $\mathbb{P}\left(\left\|\frac{1}{\alpha}\boldsymbol{Z}_5\right\|_2 \leq \frac{b^2}{\alpha(1-b^2)}\sqrt{c_{\text{new}}r}\gamma_{\text{new}}\gamma\right) = 1$

Combining the above yields the claim of Lemma 3.6.3.

⊠

3.8 Alternative Subspace Model and Algorithm

3.8.1 Deletion Model

Recall Signal Model 3.2.8 which assumes

$$\boldsymbol{P}_t = \begin{cases} [\boldsymbol{P}_{t-1}\boldsymbol{R}_t \setminus \boldsymbol{P}_{t,\text{old}}\ \boldsymbol{P}_{t,\text{new}}] & \text{if } t = t_1 \text{ or } t_2 \text{ or } \ldots t_J \\ \boldsymbol{P}_{t-1} & \text{otherwise} \end{cases} \tag{3.24}$$

According to this model, at the change times t_j, some directions in the span of $\boldsymbol{P}_{t-1}$ may be removed, and new directions may be added.

In this section we will intoduce an extension of Algorithm 3 which deletes directions from the estimate of span($\boldsymbol{P}_t$) by re-estimating the previous subspace before begining to estimate new directions by the same projection PCA procedure. We prove a result similar to Theorem 3.2.15 for this algorithm.

Definition 3.8.1. *Define*

- $c_{\text{new}} := \max_j c_{j,\text{new}}$
- $c_{\text{dif}} := \max_j \sum_{i=1}^{j}(c_{j,\text{new}} - c_{j,\text{old}})$
- $r_j := \text{rank}(\boldsymbol{P}_{(j)})$
- $r_{\max} := \max_j r_j = r_0 + c_{\text{dif}}$
- $r = r_0 + Jc_{\text{new}}$

In order to generate a new estimate of the subspace, we need a clustering assumption on the eigenvalues of $\boldsymbol{\Lambda}_{a,t}$ after the subspace change has stabilized. The reason for the cluster-PCA algorithm and clustering assumption is that the condition number of $\boldsymbol{\Lambda}_{a,t}$ may be large, and the error in the PCA step, $\boldsymbol{e}_t$, is correlated with the true data $\boldsymbol{\ell}_t$ [12].

Model 3.8.2. *Assume Signal Model 3.2.8 and assume:*

1. *During the interval* $[t_{j+1} - d_2, t_{j+1} - 1]$, $\mathbf{\Lambda}_{a,t}$ *is constant*

2. *There exists a partition* $\mathcal{G}_{j,1}, \mathcal{G}_{j,2}, \ldots, \mathcal{G}_{j,\vartheta}$ *of the index set* $\{1, 2, \ldots, r_j\}$ *with* $\min_{i \in \mathcal{G}_{j,k}} \lambda_i(\mathbf{\Lambda}_{a,t_{j+1}-1}) > \max_{i \in \mathcal{G}_{j,k+1}} \lambda_i(\mathbf{\Lambda}_{a,t_{j+1}-1})$ *for* $k = 1, \ldots, \vartheta - 1$ *that satisfies*

$$\tilde{g} \leq \tilde{g}^+ \quad \textit{and} \quad \tilde{\chi} \leq \tilde{\chi}^+ \quad \textit{and} \quad \vartheta \leq \vartheta^+$$

for a $\tilde{g}^+ > 1$ *but not too large and a* $\tilde{\chi}^+ < 1$. *Where*

$$\tilde{g} := \max_{j,k} \frac{\max\limits_{i \in \mathcal{G}_{j,k}} \lambda_i(\mathbf{\Lambda}_{a,t_{j+1}-d_2})}{\min\limits_{i \in \mathcal{G}_{j,k}} \lambda_i(\mathbf{\Lambda}_{a,t_{j+1}-d_2})} \quad \textit{and } \tilde{\chi} := \max_{j,k} \frac{\max\limits_{i \in \mathcal{G}_{j,k+1}} \lambda_i(\mathbf{\Lambda}_{a,t_{j+1}-d_2})}{\min\limits_{i \in \mathcal{G}_{j,k}} \lambda_i(\mathbf{\Lambda}_{a,t_{j+1}-d_2})}.$$

Notice that $\tilde{g} \geq 1$ *and measures how close the eigenvalues within a cluster are, and* $\tilde{\chi} \leq 1$ *and measures how far apart the adjacent clusters are.*

Definition 3.8.3. *Define*

1. $\boldsymbol{G}_{j,k} := (\boldsymbol{P}_{(j)})_{\mathcal{G}_{j,k}}$ *to be the eigenvectors corresponding to the eigenvalues indexed by* $\mathcal{G}_{j,k}$, *so* $\text{range}(\boldsymbol{P}_{(j)}) = \text{range}([\boldsymbol{G}_{j,1} \ \boldsymbol{G}_{j,2} \ \ldots \ \boldsymbol{G}_{j,\vartheta}])$;

2. $\tilde{c}_{j,k} := |\mathcal{G}_{j,k}| = \text{rank}(\boldsymbol{G}_{j,k})$, *so* $\sum_{k=1}^{\vartheta} \tilde{c}_{j,k} = r_j$;

3. $\lambda^-_{j,k} := \min_{i \in \mathcal{G}_{j,k}} \min_{t \in [t_{j+1}-d_2, t_{j+1}-1]} \lambda_i(\mathbf{\Lambda}_{a,t})$ *and* $\lambda^+_{j,k} := \max_{i \in \mathcal{G}_{j,k}} \max_{t \in [t_{j+1}-d_2, t_{j+1}-1]} \lambda_i(\mathbf{\Lambda}_{a,t})$

4. $\tilde{g}_{j,k} := \frac{\lambda^+_{j,k}}{\lambda^-_{j,k}}$ *and* $\tilde{\chi}_{j,k} := \frac{\lambda^+_{j,k+1}}{\lambda^-_{j,k}}$ *(notice* $\tilde{g} = \max_{j,k} \tilde{g}_{j,k}$ *and* $\tilde{\chi} = \max_{j,k} \tilde{h}_{j,k}$*)*

3.8.2 Performance Guarantee for Algorithm 4

Theorem 3.8.4 (Correctness result for Algorithm 4 under Signal Model 3.8.2)**.** *Pick a* ζ *that satisfies*

$$\zeta \leq \min \left\{ \frac{10^{-4}}{(r_{\max} + c_{\text{new}})^2}, \frac{1.5 \times 10^{-4}}{(r_{\max} + c_{\text{new}})^2 f}, \frac{1}{(r_{\max} + c_{\text{new}})^3 \gamma^2}, \frac{0.01\lambda^-}{b^2 (r_{\max} + c_{\text{new}})^3 \gamma^2}, \gamma \right\}.$$

Suppose

1. $\|(\boldsymbol{I} - \hat{\boldsymbol{P}}_{(0)} \hat{\boldsymbol{P}}_{(0)}{}')\boldsymbol{P}_{(0)}\|_2 \leq r_0 \zeta$;

2. *The algorithm parameters are set as:*

   - $K = \left\lceil \frac{\log(0.16 c_{\text{new}} \zeta)}{\log(0.4)} \right\rceil$;
   - $\xi = \sqrt{c_{\text{new}}}\gamma_{\text{new}} + (\sqrt{r} + \sqrt{c_{\text{new}}})\sqrt{\zeta}$;
   - $\omega = 7\xi$;
   - $\alpha = C_1(\log(6KJ) + 11\log(n))$ *for a constant* $C_1 \geq C_{\text{add}}$ *with*

     $$C_{\text{add}} := \frac{4800}{(\zeta\lambda^-)^2} \max\{16, (1.2\xi)^4\}$$

   - $\tilde{\alpha} = C_2(\log(50\vartheta J) + 11\log(n))$ *for a* $C_2 \geq C_{\text{del}}$ *with*

     $$C_{\text{del}} := \frac{8 \cdot 100^2 \cdot 4^2 \cdot r^2\gamma^4}{(c_{\text{new}}\zeta\lambda^-)^2}$$

3. *Signal Model 3.8.2 holds with* $b \leq 0.1$ *and*

   - $d_1 \geq (K+2)\alpha$ *and* $d_2 \geq \vartheta^+\tilde{\alpha} + 2\alpha$;
   - $t_{j+1} - t_j \geq d_1 + d_2$ *for all* j;
   - $\sqrt{c_{\text{new}}}\gamma_{\text{new}} + (\sqrt{r} + \sqrt{c_{\text{new}}})\sqrt{\zeta} \leq \frac{x_{\min}}{14}$;
   - $g \leq \sqrt{2}$;
   - $b^2 c_{\text{new}}\eta_{\text{new}} g \leq 0.5$ *(as before, because* $b \leq 0.1$*, this will be satisfied if* $c_{\text{new}}\eta_{\text{new}} g \leq 50$ *)*

4. *The clustering assumption holds with* $\tilde{g}^+ = 1.5$, $\tilde{\chi}^+ = 0.2$, *and* $\vartheta^+ = 3$.

5. *The support of* $\boldsymbol{x}_t$ *changes enough so that for* α *and* $\tilde{\alpha}$ *as chosen above, Signal Model 3.2.11 holds with* $\beta = h^+ \min\{\alpha, \tilde{\alpha}\}$ *and*

   $$\lceil \varrho \rceil^2 h^+ \leq 0.0024, \text{ and } \varrho_2 s \max\{\alpha, \tilde{\alpha}\} \leq n$$

   or Signal Model 3.2.12 holds with $s \leq (6 \times 10^{-4}) \min\{\alpha, \tilde{\alpha}\}$ *and* $\max\{\alpha, \tilde{\alpha}\} \leq \frac{n}{m}$.

6. *The low dimensional subspace is dense such that*

   - $\max_j \kappa_{2s}(\boldsymbol{P}_{(j)}) \leq 0.3$;
   - $\max_j \kappa_{2s}(\boldsymbol{P}_{(j),\text{new}}) \leq 0.02$.

Then, with probability at least $1 - 2n^{-10}$*, at all times* t

1. *The support of* $\boldsymbol{x}_t$ *is recovered exactly, i.e.* $\hat{\mathcal{T}}_t = \mathcal{T}_t$
2. *The estimate of the subspace change time satisfies* $t_j \le \hat{t}_j \le t_j + 2\alpha$*, for* $j = 1, \dots, J$*;*
3. *The estimate of the number of new directions is correct, i.e.* $\hat{c}_{j,\text{new},k} = c_{j,\text{new}}$ *for* $j = 1, \dots, J$ *and* $k = 1, \dots, K$*;*
4. *The recovery error satisfies:*

$$\|\hat{\boldsymbol{x}}_t - \boldsymbol{x}_t\|_2 \le \begin{cases} 1.2(\sqrt{\zeta} + \sqrt{c_{\text{new}}}\gamma_{\text{new}}) & t \in [t_j, \hat{t}_j] \\ 1.2\left(1.84\sqrt{\zeta} + (0.4)^{k-1}\sqrt{c_{\text{new}}}\gamma_{\text{new}}\right) & t \in [\hat{t}_j + (k-1)\alpha, \hat{t}_j + k\alpha - 1], \\ & k = 1, 2, \dots, K \\ 2.4\sqrt{\zeta} & t \in [\hat{t}_j + K\alpha, t_{j+1} - 1] \end{cases}$$

5. *For* $j = 1, \dots, J$*,* $t_j \le \hat{t}_j \le t_j + 2\alpha$*.*
6. *The subspace error* $\text{SE}_t := \|(\boldsymbol{I} - \hat{\boldsymbol{P}}_t \hat{\boldsymbol{P}}_t')\boldsymbol{P}_t\|_2$ *satisfies:*

$$\text{SE}_t \le \begin{cases} 1 & t \in [t_j, \hat{t}_j] \\ 10^{-2}\sqrt{\zeta} + 0.4^{k-1} & t \in [\hat{t}_j + (k-1)\alpha, \hat{t}_j + k\alpha - 1],\ k = 1, 2, \dots, K \\ 10^{-2}\sqrt{\zeta} & t \in [\hat{t}_j + K\alpha, t_{j+1} - 1]. \end{cases}$$

3.8.3 Discussion

First we point out a significant limitation of the above result. Theorem 3.8.4 and Algorithm 4 assume that the clusters $\mathcal{G}_{j,k}$ are known a priori. This is not a very realistic assumption, as determining the clusters would require knowledge of all the eigenvalues of the true covariance matrix $\boldsymbol{\Lambda}_{a,t}$.

Let us compare the result for ReProCS (Algorithm 3) with that for ReProCS-cPCA (Algorithm 4) for the more general subspace change model. The ReProCS result needs $\kappa_{2s}([\boldsymbol{P}_0, \boldsymbol{P}_{1,\text{new}}, \dots, \boldsymbol{P}_{J,\text{new}}]) \le 0.3$ while ReProCS-cPCA only needs $\max_j \kappa_{2s}(\boldsymbol{P}_{(j)}) \le 0.3$

and $\max_j \kappa_{2s}(\boldsymbol{P}_{(j),\text{new}}) \leq 0.02$. Recall that $\boldsymbol{L} := [\boldsymbol{\ell}_1, \boldsymbol{\ell}_2, \dots, \boldsymbol{\ell}_{t_{\max}}]$, $\boldsymbol{X} := [\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_{t_{\max}}]$, $r_{\text{mat}} := \text{rank}(\boldsymbol{L})$ and $s_{\text{mat}} = |\text{support}(\boldsymbol{X})|$. Also recall that in our model $s_{\text{mat}} \leq s t_{\max}$.

Clearly $\text{rank}([\boldsymbol{P}_0, \boldsymbol{P}_{1,\text{new}}, \dots, \boldsymbol{P}_{J,\text{new}}]) = r_{\text{mat}}$, and so $r_{\text{mat}} \leq r_0 + J c_{\text{new}}$. Thus if we assume $\kappa_1([\boldsymbol{P}_0, \boldsymbol{P}_{1,\text{new}}, \dots, \boldsymbol{P}_{J,\text{new}}])^2 \leq \mu r_{\text{mat}}/n$, and $\kappa_1(\boldsymbol{P}_{j,\text{new}}) \leq \tilde{\mu} c_{\text{new}}/n$ then ReProCS needs

$$2\mu s r_{\text{mat}} \leq 0.09n \text{ and } 2\tilde{\mu} s c_{\text{new}} \leq 0.0004n$$

where $r_{\text{mat}} \leq r_0 + J c_{\text{new}}$. The support change model (Signal Model 3.2.11) requires $s \in \mathcal{O}(\frac{n}{\log n})$ and $J \in \mathcal{O}(n)$. Thus, if s grows as $n/\log n$, then r_{mat} can only grow as $\log n$. Thus, ReProCS allows

$$s_{\text{mat}} \in \mathcal{O}\left(\frac{n t_{\max}}{\log n}\right) \text{ and } r_{\text{mat}} \in \mathcal{O}(\log n)$$

As explained earlier this is a stronger requirement than PCP which allows

$$s_{\text{mat}} \in \mathcal{O}(n t_{\max}) \text{ and } r_{\text{mat}} \in \mathcal{O}\left(\frac{n}{(\log n)^2}\right)$$

On the other hand, using an argument similar to the one above, ReProCS-cPCA only needs

$$2\mu s(r_0 + c_{\text{dif}}) \leq 0.09n \text{ and } 2\tilde{\mu} s c_{\text{new}} \leq 0.0004n.$$

The support change model (Signal Model 3.2.11) requires $s \in \mathcal{O}(\frac{n}{\log n})$ and $J \in \mathcal{O}(n)$. Thus if s grows as $n/\log n$, r_0 can grow at most as $\log n$; however, c_{new} needs to be constant because of the lower bound on $x_{\min}$. Since $r_{\text{mat}} \leq r_0 + J c_{\text{new}}$, this means r_{mat} can grow linearly with n. Thus, ReProCS-cPCA allows

$$s_{\text{mat}} \in \mathcal{O}\left(\frac{n t_{\max}}{\log n}\right) \text{ and } r_{\text{mat}} \in \mathcal{O}(n)$$

This requirement is comparable to what PCP needs: the requirement on s_{mat} is slightly stronger than PCP while that on r_{mat} is slightly weaker.

3.8.4 Cluster-PCA Algorithm (from [12])

In this section we present the ReProCS algorithm with cluster-PCA as Algorithm 4. It performs the same subspace change detection and estimation of new directions as ReProCS, but it differs in that at each subspace change time it performs a re-estimation of the previous subspace that effectively removes old directions from the estimate of $\boldsymbol{P}_t$.

One way to re-estimate the current subspace would be by standard PCA: at $t = \hat{t}_{j+1}$, compute

$$\hat{\boldsymbol{P}}_t \leftarrow \text{eigenvectors}\left(\sum_{\tau=\hat{t}_{j+1}-2\alpha-\tilde{\alpha}}^{\hat{t}_{j+1}-2\alpha} \hat{\boldsymbol{\ell}}_\tau \hat{\boldsymbol{\ell}}_\tau{}'\right)$$

Using the same proof method as used for Theorem 3.2.15, it can be shown that, as long as $f = \frac{\lambda^+}{\lambda^-}$ is small enough, we can show that this will give an accurate estimate of range($\boldsymbol{P}_{(j)}$). However f cannot be small because our problem definition allows large noise, $\boldsymbol{\ell}_t$, but assumes slow subspace change. To resolve this problem, we recover the eigenvectors in clusters where the ratio of the largest to smallest eigenvalue in each cluster is small. We also have to assume that these clusters are sufficiently far apart from each other.

In a process that we call cluster-PCA, Algorithm 4 sequentially recovers eigenvectors corresponding to groups of eigenvalues that are close to each other and separated from the rest of the eigenvalues. The eigenvectors corresponding to the largest eigenvalues are recovered first. After this, the data is projected perpendicular to the estimated subspace to recover the eigenvectors corresponding to the next largest cluster. This process continues until all of span($\boldsymbol{P}_{(j)}$) has been estimated. All of this takes place at $t = \hat{t}_{j+1}$ using the estimates $\hat{\boldsymbol{\ell}}_t$ from the intervals $[(\hat{t}_{j+1} - \vartheta\tilde{\alpha} - 2\alpha) + (k-1)\tilde{\alpha} + 1, (\hat{t}_{j+1} - \vartheta\tilde{\alpha} - 2\alpha) + k\tilde{\alpha}]$, $k = 1, \ldots, \vartheta$.

The motivation for the cluster-PCA step is Signal Model 3.2.8 where at the times when new directions are added to the subspace, some may also be removed. This is a more appropriate model when the sparse signal $\boldsymbol{x}_t$ is the signal of interest. Under this model we are able to relax the requirements needed to prove a similar performance guarantee to Theorem 3.2.15. The difference is that instead of needing span($[\boldsymbol{\ell}_1, \ldots, \boldsymbol{\ell}_{t_{\max}}]$) to be dense, we only require span($\boldsymbol{P}_t$) to be dense for all t. Of course when no directions are removed from the subspace, these are equivalent, because span($\boldsymbol{P}_t$) $\subseteq$ span($\boldsymbol{P}_{t+1}$) for all t.

3.8.5 Proof of Theorem 3.8.4

Definition 3.8.5. *Redefine $\hat{\boldsymbol{P}}_{(j),*}$ to be the estimate of $\boldsymbol{P}_{(j),*} = \boldsymbol{P}_{(j-1)}$ after the cluster PCA step. That is $\hat{\boldsymbol{P}}_{(j),*} := \hat{\boldsymbol{P}}_{\hat{t}_j}$. Previously we had defined $\hat{\boldsymbol{P}}_{(j),*}$ to be the estimate of $\boldsymbol{P}_{(j),*}$ after the final projection PCA (addition) step, so this new definition is a natural extension. In both*

Algorithm 4 Recursive Projected CS with cluster-PCA (ReProCS-cPCA)

Parameters: algorithm parameters: ξ, ω, α, $\tilde{\alpha}$, K, model parameters: λ^-, ϑ, and $\tilde{c}_{j,k}$
Input: $n \times 1$ vector, $\boldsymbol{m}_t$, and $n \times r_0$ basis matrix $\hat{\boldsymbol{P}}_{(0)}$.
Output: $n \times 1$ vectors $\hat{\boldsymbol{x}}_t$ and $\hat{\boldsymbol{\ell}}_t$, a basis matrix $\hat{\boldsymbol{P}}_t$, $\hat{t}_j$, $\hat{c}_{j,\text{new},k}$.
In step 4(c)ii of Algorithm 3, include the following cluster-PCA (cPCA) step.
Re-estimate range($\boldsymbol{P}_{(j-1)}$) by cluster-PCA (in order to remove the deleted directions)

1. set $\hat{\boldsymbol{G}}_{j-1,0} \leftarrow [.]$
2. For $i = 1, 2, \cdots, \vartheta$,
   - let $\hat{\boldsymbol{G}}_{j-1,\text{det},i} := [\hat{\boldsymbol{G}}_{j-1,1}, \hat{\boldsymbol{G}}_{j-1,2}, \ldots \hat{\boldsymbol{G}}_{j-1,i-1}]$ and compute

$$\boldsymbol{\mathcal{M}}_{cpca} = (\boldsymbol{I} - \hat{\boldsymbol{G}}_{j-1,\text{det},i}\hat{\boldsymbol{G}}_{j-1,\text{det},i}{}') \left(\frac{1}{\tilde{\alpha}} \sum_{t \in \mathcal{I}_{j,k}} \hat{\boldsymbol{\ell}}_\tau \hat{\boldsymbol{\ell}}_\tau{}' \right) (\boldsymbol{I} - \hat{\boldsymbol{G}}_{j-1,\text{det},i}\hat{\boldsymbol{G}}_{j-1,\text{det},i}{}')$$

   - $\hat{\boldsymbol{G}}_{j-1,i} \leftarrow \text{eigenvectors}(\boldsymbol{\mathcal{M}}_{cpca}, , \tilde{c}_{j,i})$

   End for
3. set $\hat{\boldsymbol{P}}_t \leftarrow [\hat{\boldsymbol{G}}_{j-1,1} \cdots \hat{\boldsymbol{G}}_{j-1,\vartheta}]$, $\hat{\boldsymbol{P}}_{*,t} \leftarrow \hat{\boldsymbol{P}}_t$, and $\hat{\boldsymbol{P}}_{t,\text{new}} \leftarrow [.]$.

The function eigenvectors($\boldsymbol{\mathcal{M}}, , c$) returns the matrix containing the eigenvectors corresponding to the c largest eigenvalues.

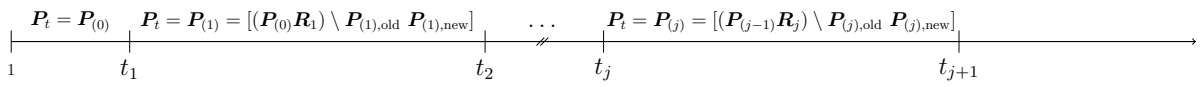

Figure 3.7 Signal Model 3.2.8

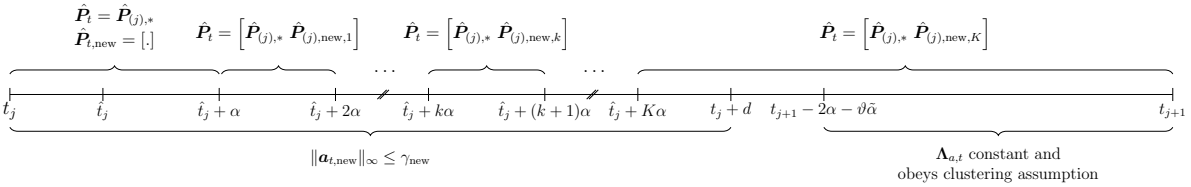

Figure 3.8 Algorithm 4

Figure 3.9: Diagrams for Signal Model 3.2.8 and Algorithm 4

cases, $\hat{\boldsymbol{P}}_{(j),*}$ *is the final estimate of* $\boldsymbol{P}_{(j-1)}$.

With the new definition of $\hat{\boldsymbol{P}}_{(j),*}$*, the statement of the definition of* $\zeta_{j,*}$ *remains the same:*

$$\zeta_{j,*} := \text{dif}(\hat{\boldsymbol{P}}_{(j),*}, \boldsymbol{P}_{(j),*})$$

The tilde ($\tilde{}$) will be used to indicate objects in the cluster PCA proof that have corresponding parts in the projection PCA proof.

Definition 3.8.6. *Redefine*

1. $\zeta_*^+ := r_{\max}\zeta$
2. $\tilde{\zeta}_k^+ := \dfrac{b_{\tilde{\boldsymbol{\mathcal{H}}},k}}{b_{\tilde{\boldsymbol{A}},k} - b_{\tilde{\boldsymbol{A}},k,\perp} - b_{\tilde{\boldsymbol{\mathcal{H}}},k}}$

Here $b_{\tilde{\boldsymbol{\mathcal{H}}},k} := \tilde{b}_2 + \tilde{b}_{4,k} + \tilde{b}_{6,k}$. *Where,*

$$\begin{aligned}
\tilde{b}_2 &:= \rho^2 h^+ (\phi^+)^2 (r_{\max} + c_{\text{new}})^2 \zeta^2 \big(b^2 r\gamma^2 + (1-b^2)\lambda^+\big) + \epsilon \\
\tilde{b}_{4,k} &:= (r\zeta)\left(\frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + (1-b^2)\lambda^+ + \epsilon\right)(r_{\max} + c_{\text{new}})\zeta\left(\sqrt{\rho^2 h^+}\phi^+\right) + \\
&\qquad \left(\frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + (1-b^2)\lambda^+_{j,k} + \epsilon\right)(r_{\max} + c_{\text{new}})\zeta\left(\sqrt{\rho^2 h^+}\phi^+\right) \\
\tilde{b}_{6,k} &:= r\zeta\left(\frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + 4\epsilon\right) + \frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + 4\epsilon + (r\zeta)^2\left(\frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + (1-b^2)\lambda^+_{j,k-1} + \epsilon\right) + \\
&\qquad r\zeta\left(\frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + 4\epsilon\right) + \frac{(r\zeta)^3}{\sqrt{1-r^2\zeta^2}}\left(\frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + 4\epsilon\right) + \\
&\qquad \frac{(r\zeta)^2}{\sqrt{1-r^2\zeta^2}}\left(\frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + (1-b^2)\lambda^+_{j,k+1} + \epsilon\right).
\end{aligned}$$

Also, let

$$\begin{aligned}
b_{\tilde{\boldsymbol{A}},k} &:= (1-r^2\zeta^2)\left[(1-b^2)\lambda^-_k - \epsilon\right] - \sqrt{1-r^2\zeta^2}\left(\frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + 4\epsilon\right)\left(r\zeta + \frac{r^2\zeta^2}{\sqrt{1-r^2\zeta^2}}\right) \\
b_{\tilde{\boldsymbol{A}},k,\perp} &:= r^2\zeta^2\left(\frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + (1-b^2)\lambda^+_{j,k-1} + \epsilon\right) - 2r\zeta\left(\frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + (1-b^2)\lambda^+_{j,k} + \epsilon\right) - \\
&\qquad \left(\frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + (1-b^2)\lambda^+_{j,k+1} + \epsilon\right).
\end{aligned}$$

To prove Theorem 3.8.4 we need to define a new event that says the $(j,k)^{\text{th}}$ cluster PCA step was successful.

Definition 3.8.7. *Define*

$$\text{CPCA}^a_{j,k} := \left\{ \left\| \left(\boldsymbol{I} - \sum_{i=1}^{k} \hat{\boldsymbol{G}}_{j,i} \hat{\boldsymbol{G}}_{j,i}{}' \right) \boldsymbol{G}_{j,k} \right\|_2 \leq \tilde{\zeta}^+_{j,k} \right\}$$

and redefine

$$\Gamma^a_{j,0} := \Gamma_{j-1,\text{end}} \cap \text{DET}^a \cap \text{CPCA}^a_{j-1,1} \cap \cdots \cap \text{CPCA}^a_{j-1,\vartheta} \quad \textit{for } a = u_j \textit{ or } a = u_j + 1$$

The definitions,

$$\begin{aligned}
\Gamma^a_{j,k} &:= \Gamma^a_{j,k-1} \cap \text{PPCA}^a_{j,k} && \textit{for } a = u_j \textit{ or } a = u_j + 1 \\
\Gamma_{j,\text{end}} &:= \left(\Gamma^{u_j}_{j,K} \cap \text{NODETS}^{u_j}_j\right) \cup \left(\Gamma^{u_j+1}_{j,K} \cap \text{NODETS}^{u_j+1}_j\right)
\end{aligned}$$

remain the same.

Lemma 3.8.8.

$$\mathbb{P}\left(\mathrm{CPCA}^a_{j,k} \mid \Gamma_{j-1,\mathrm{end}}, \mathrm{DET}^a_j, \mathrm{CPCA}^a_{j,1}, \ldots, \mathrm{CPCA}^a_{j,k-1}\right) \geq p_{cpca}$$

for $a = u_{j+1}$ *or* $a = u_{j+1} + 1$.

Fact 3.8.9.

1. *If* $\zeta_{j,*} \leq \zeta^+_{j,*}$ *and* $\zeta_{j,\mathrm{new},k} \leq \zeta^+_{k,\mathrm{new}}$, *for* $k = 1, \ldots, K$ *then* $\mathrm{dif}(\hat{\boldsymbol{P}}_{\hat{t}_j + K\alpha}, \boldsymbol{P}_{(j)}) \leq \zeta^+_* + c_{\mathrm{new}}\zeta$
2. *If* $\tilde{\zeta}_{j,k} \leq \tilde{\zeta}^+_k$ *for* $k = 1, \ldots, \vartheta$, *then* $\zeta_{j+1,*} := \mathrm{dif}(\hat{\boldsymbol{P}}_{(j+1),*}, \boldsymbol{P}_{(j+1),*}) = \mathrm{dif}(\hat{\boldsymbol{P}}_{\hat{t}_{j+1}}, \boldsymbol{P}_{(j)}) \leq r_{\max}\zeta = \zeta^+_*$.
3. *Thus the event* $\Gamma_{j,\mathrm{end}}$ *implies* $\mathrm{dif}(\hat{\boldsymbol{P}}_{\hat{t}_j + K\alpha}, \boldsymbol{P}_{(j)}) \leq \zeta^+_* + c_{\mathrm{new}}\zeta$. *The event* $\Gamma^a_{j,0}$ *implies* $\zeta_{j,*} \leq \zeta^+_* = r_{\max}\zeta$ *for* $a = u_j$ *or* $a = u_{j+1}$. *Thus the event* $\Gamma^a_{j,k-1}$ *also implies this.*

Corollary 3.8.10. *Combining Lemmas 3.5.11, 3.5.12, 3.5.14, and 3.8.8 gives*

$$\begin{aligned}
\mathbb{P}\left(\Gamma_{j,\mathrm{end}} \mid \Gamma_{j-1,\mathrm{end}}\right) &= \mathbb{P}\Bigg(\left(\mathrm{DET}^{u_j} \bigcap_{k=1}^{\vartheta} \mathrm{CPCA}^{u_j}_{j-1,k} \bigcap_{k=1}^{K} \mathrm{PPCA}^{u_j}_{j,k}\right) \cup \\
&\qquad \left(\overline{\mathrm{DET}^{u_j}} \cap \mathrm{DET}^{u_j+1} \bigcap_{k=1}^{\vartheta} \mathrm{CPCA}^{u_j+1}_{j-1,k} \bigcap_{k=1}^{K} \mathrm{PPCA}^{u_j+1}_{j,k}\right) \mid \Gamma_{j-1,\mathrm{end}}\Bigg) \\
&= \mathbb{P}\left(\mathrm{DET}^{u_j} \bigcap_{k=1}^{\vartheta} \mathrm{CPCA}^{u_j}_{j-1,k} \bigcap_{k=1}^{K} \mathrm{PPCA}^{u_j}_{j,k} \mid \Gamma_{j-1,\mathrm{end}}\right) \\
&\qquad + \mathbb{P}\left(\overline{\mathrm{DET}^{u_j}} \cap \mathrm{DET}^{u_j+1} \bigcap_{k=1}^{\vartheta} \mathrm{CPCA}^{u_j+1}_{j-1,k} \bigcap_{k=1}^{K} \mathrm{PPCA}^{u_j+1}_{j,k} \mid \Gamma_{j-1,\mathrm{end}}\right) \\
&\geq p_{\mathrm{det},0} \cdot (p_{\mathrm{cpca}})^{\vartheta} \cdot (p_{\mathrm{ppca}})^K + (1 - p_{\mathrm{det},0}) \cdot p_{\mathrm{det},1} \cdot (p_{\mathrm{cpca}})^{\vartheta} \cdot (p_{\mathrm{ppca}})^K \\
&\geq p_{\mathrm{det},1} \cdot (p_{\mathrm{cpca}})^{\vartheta} \cdot (p_{\mathrm{ppca}})^K
\end{aligned}$$

3.8.6 Proof of Lemma 3.8.8

Definition 3.8.11. *Define*

$$\tilde{\mathcal{I}}_{j,k} := \left[(\hat{t}_{j+1} - \vartheta\tilde{\alpha} - 2\alpha) + (k-1)\tilde{\alpha} + 1, (\hat{t}_{j+1} - \vartheta\tilde{\alpha} - 2\alpha) + k\tilde{\alpha}\right]$$

Definition 3.8.12. *Define*

$$\tilde{X}_{j,k} := [\boldsymbol{\nu}_1, \ldots, \boldsymbol{\nu}_{(\hat{t}_{j+1} - \vartheta\tilde{\alpha} - 2\alpha) + k\tilde{\alpha}}]$$

and

$$\tilde{\Gamma}_{j,k}^{\hat{u}_j} := \Gamma_{j-1,\text{end}} \cap \text{DET}^{\hat{u}_j} \cap \text{CPACA}_{j-1,1}^{\hat{u}_j} \cap \cdots \cap \text{CPCA}_{j-1,k}^{\hat{u}_j}$$

for $\hat{u}_j = u_j$ *or* $\hat{u}_j = u_j + 1$.

Definition 3.8.13.

1. *Let* $\tilde{\boldsymbol{D}}_{j,k,\text{cur}} \overset{\text{QR}}{=} \tilde{\boldsymbol{E}}_{j,k,\text{cur}} \tilde{\boldsymbol{R}}_{j,k,\text{cur}}$ *denote its reduced QR decomposition. So* $\tilde{\boldsymbol{E}}_{j,k,\text{cur}}$ *is a basis matrix, and* $\tilde{\boldsymbol{R}}_{j,k,\text{cur}}$ *is upper triangular. Let* $\tilde{\boldsymbol{E}}_{j,k,\text{cur},\perp}$ *be a basis matrix for the orthogonal complement of* $\text{range}(\tilde{\boldsymbol{E}}_{j,k,\text{cur}})$.

2. *Using* $\tilde{\boldsymbol{E}}_{j,k,\text{cur}}$ *and* $\tilde{\boldsymbol{E}}_{j,k,\text{cur},\perp}$ *, define*

$$\tilde{\boldsymbol{A}}_{j,k} := \frac{1}{\tilde{\alpha}} \sum_{t \in \tilde{\mathcal{I}}_{j,k}} \tilde{\boldsymbol{E}}_{j,k,\text{cur}}{}' \left(\boldsymbol{I} - \hat{\boldsymbol{G}}_{j,k,\text{det}} \hat{\boldsymbol{G}}_{j,k,\text{det}}{}' \right) \boldsymbol{\ell}_t \boldsymbol{\ell}_t{}' \left(\boldsymbol{I} - \hat{\boldsymbol{G}}_{j,k,\text{det}} \hat{\boldsymbol{G}}_{j,k,\text{det}}{}' \right) \tilde{\boldsymbol{E}}_{j,k,\text{cur}}$$

$$\tilde{\boldsymbol{A}}_{j,k,\perp} := \frac{1}{\tilde{\alpha}} \sum_{t \in \tilde{\mathcal{I}}_{j,k}} \tilde{\boldsymbol{E}}_{j,k,\text{cur},\perp}{}' \left(\boldsymbol{I} - \hat{\boldsymbol{G}}_{j,k,\text{det}} \hat{\boldsymbol{G}}_{j,k,\text{det}}{}' \right) \boldsymbol{\ell}_t \boldsymbol{\ell}_t{}' \left(\boldsymbol{I} - \hat{\boldsymbol{G}}_{j,k,\text{det}} \hat{\boldsymbol{G}}_{j,k,\text{det}}{}' \right) \tilde{\boldsymbol{E}}_{j,k,\text{cur},\perp}$$

and let

$$\tilde{\boldsymbol{\mathcal{A}}}_{j,k} := \begin{bmatrix} \tilde{\boldsymbol{E}}_{j,k,\text{cur}} & \tilde{\boldsymbol{E}}_{j,k,\text{cur},\perp} \end{bmatrix} \begin{bmatrix} \tilde{\boldsymbol{A}}_{j,k} & \boldsymbol{0} \\ \boldsymbol{0} & \tilde{\boldsymbol{A}}_{k,\perp} \end{bmatrix} \begin{bmatrix} \tilde{\boldsymbol{E}}_{j,k,\text{cur}}{}' \\ \tilde{\boldsymbol{E}}_{j,k,\text{cur},\perp}{}' \end{bmatrix}$$

3. *Define*

$$\tilde{\boldsymbol{\mathcal{H}}}_{j,k} = \frac{1}{\tilde{\alpha}} \sum_{t \in \tilde{\mathcal{I}}_{j,k}} \left(\boldsymbol{I} - \hat{\boldsymbol{G}}_{j,k,\text{det}} \hat{\boldsymbol{G}}_{j,k,\text{det}}{}' \right) \hat{\boldsymbol{\ell}}_t \hat{\boldsymbol{\ell}}_t{}' \left(\boldsymbol{I} - \hat{\boldsymbol{G}}_{j,k,\text{det}} \hat{\boldsymbol{G}}_{j,k,\text{det}}{}' \right) - \tilde{\boldsymbol{\mathcal{A}}}_{j,k}$$

From Algorithm 4,

$$\frac{1}{\tilde{\alpha}} \sum_{t \in \tilde{\mathcal{I}}_{j,k}} \left(\boldsymbol{I} - \hat{\boldsymbol{G}}_{j,k,\text{det}} \hat{\boldsymbol{G}}_{j,k,\text{det}}{}' \right) \hat{\boldsymbol{\ell}}_t \hat{\boldsymbol{\ell}}_t{}' \left(\boldsymbol{I} - \hat{\boldsymbol{G}}_{j,k,\text{det}} \hat{\boldsymbol{G}}_{j,k,\text{det}}{}' \right)$$

$$\overset{\text{EVD}}{=} \begin{bmatrix} \hat{\boldsymbol{G}}_{j,k} & \hat{\boldsymbol{G}}_{j,k,\perp} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\Lambda}}_{j,k} & \boldsymbol{0} \\ \boldsymbol{0} & \hat{\boldsymbol{\Lambda}}_{j,k,\perp} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{G}}_{j,k}{}' \\ \hat{\boldsymbol{G}}_{j,k,\perp}{}' \end{bmatrix}.$$

Define $\tilde{\zeta}_{j,k} := \text{dif}([\hat{\boldsymbol{G}}_{j,1}, \ldots, \hat{\boldsymbol{G}}_{j,k}], \boldsymbol{G}_{j,k})$. Recall Lemma 3.5.17, which for the matrices above says that if $\lambda_{\min}(\tilde{\boldsymbol{A}}_{j,k}) - \|\tilde{\boldsymbol{A}}_{j,k,\perp}\|_2 - \|\tilde{\boldsymbol{\mathcal{H}}}_{j,k}\|_2 > 0$, then

$$\tilde{\zeta}_{j,k} \leq \frac{\|\tilde{\boldsymbol{\mathcal{H}}}_{j,k}\|_2}{\lambda_{\min}(\tilde{\boldsymbol{A}}_{j,k}) - \|\tilde{\boldsymbol{A}}_{j,k,\perp}\|_2 - \|\tilde{\boldsymbol{\mathcal{H}}}_{j,k}\|_2} \tag{3.25}$$

Lemma 3.8.14.

$$\begin{aligned}\mathbb{P}\Bigg(\lambda_{\min}\left(\tilde{\boldsymbol{A}}_{j,k}\right) \geq & (1-r^2\zeta^2)\left[(1-b^2)\lambda_k^- - \epsilon\right] \\ & - \sqrt{1-r^2\zeta^2}\left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + 4\epsilon\right)\left(r\zeta + \frac{r^2\zeta^2}{\sqrt{1-r^2\zeta^2}}\right) \;\Big|\; \tilde{X}_{k-1}\Bigg) \\ & \geq 1 - r\digamma(\tilde{\alpha},\epsilon,c_{\text{new}}\gamma^2) - 2(3(2r)\digamma(\tilde{\alpha},\epsilon,2r\gamma^2) - (2r)\digamma(\tilde{\alpha},\epsilon,4r\gamma^2)).\end{aligned}$$

for all $\tilde{X}_{k-1} \in \tilde{\Gamma}_{k-1}$.

Lemma 3.8.15.

$$\begin{aligned}\mathbb{P}\Bigg(\lambda_{\max}\left(\tilde{\boldsymbol{A}}_{j,k,\perp}\right) \leq & r^2\zeta^2\left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + (1-b^2)\lambda_{j,k-1}^+ + \epsilon\right) \; + \\ & 2r\zeta\left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + (1-b^2)\lambda_{j,k}^+ + \epsilon\right) + \\ & \left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + (1-b^2)\lambda_{j,k+1}^+ + \epsilon\right) \;\Big|\; \tilde{X}_{k-1}\Bigg) \geq 1 - 3r\digamma(\tilde{\alpha},\epsilon,r\gamma^2)\end{aligned}$$

for all $\tilde{X}_{k-1} \in \tilde{\Gamma}_{k-1}$.

Lemma 3.8.16.

$$\begin{aligned}\mathbb{P}\Big(\|\tilde{\boldsymbol{\mathcal{H}}}_{j,k}\|_2 \leq & \tilde{b}_2 + 2\tilde{b}_{4,k} + 2\tilde{b}_{6,k} \;\Big|\; \tilde{X}_{k-1}\Big) \geq 1 - n\digamma(\tilde{\alpha},\epsilon,(\phi^+)^4\zeta^2) \\ & - r\digamma(\tilde{\alpha},\epsilon,r\gamma^2) \\ & - 3\Big(3(2r)\digamma(\tilde{\alpha},\epsilon,2r\gamma^2) - (2r)\digamma(\tilde{\alpha},\epsilon,4r\gamma^2)\Big) \\ & - 2\Big(r\digamma(\tilde{\alpha},\epsilon,c_{\text{new}}\gamma^2)\Big).\end{aligned}$$

for all $\tilde{X}_{k-1} \in \tilde{\Gamma}_{k-1}$.

Combining the above three lemmas gives Lemma 3.8.8.

3.9 Proofs of Lemmas 3.8.14, 3.8.15, and 3.8.16

3.9.1 Minor Lemmas for Proving the Main Lemmas

Definition 3.9.1. *Define*

1. $\boldsymbol{G}_{j,k,\text{det}} := [\boldsymbol{G}_{j,1}, \boldsymbol{G}_{j,2}, \ldots, \boldsymbol{G}_{j,k-1}]$, $\hat{\boldsymbol{G}}_{j,k,\text{det}} := [\hat{\boldsymbol{G}}_{j,1}, \hat{\boldsymbol{G}}_{j,2}, \ldots, \hat{\boldsymbol{G}}_{j,k-1}]$,

   $\boldsymbol{G}_{j,k,\text{undet}} := [\boldsymbol{G}_{j,k+1}, \boldsymbol{G}_{j,k+2}, \ldots, \boldsymbol{G}_{j,\vartheta}]$, $\boldsymbol{G}_{j,k,\text{cur}} := \boldsymbol{G}_{j,k}$;

2. $\boldsymbol{a}_{t,\text{det}} := \boldsymbol{G}_{\text{det},k}{}'\boldsymbol{\ell}_t$, $\boldsymbol{a}_{t,\text{cur}} := \boldsymbol{G}_{k,\text{cur}}{}'\boldsymbol{\ell}_t$, *and* $\boldsymbol{a}_{t,\text{undet}} := \boldsymbol{G}_{\text{undet},k}{}'\boldsymbol{\ell}_t$;

3. $\tilde{\boldsymbol{D}}_{j,k,\text{cur}} := (\boldsymbol{I} - \hat{\boldsymbol{G}}_{j,k,\text{det}}\hat{\boldsymbol{G}}_{j,k,\text{det}}{}')\boldsymbol{G}_{j,k}$, $\tilde{\boldsymbol{D}}_{j,k,\text{det}} := (\boldsymbol{I} - \hat{\boldsymbol{G}}_{j,k,\text{det}}\hat{\boldsymbol{G}}_{j,k,\text{det}}{}')\boldsymbol{G}_{j,k,\text{det}}$, $\tilde{\boldsymbol{D}}_{j,k,\text{undet}} := (\boldsymbol{I} - \hat{\boldsymbol{G}}_{j,k,\text{det}}\hat{\boldsymbol{G}}_{j,k,\text{det}}{}')\boldsymbol{G}_{j,k,\text{undet}}$;

Lemma 3.9.2 (Sparse Recovery for $\tilde{\mathcal{I}}_{j,k}$)**.** *Define* $\boldsymbol{\Phi}_{(j),K} := \boldsymbol{I} - \hat{\boldsymbol{P}}_{(j),*}\hat{\boldsymbol{P}}_{(j),*}{}' - \hat{\boldsymbol{P}}_{(j),\text{new},K}\hat{\boldsymbol{P}}_{(j),\text{new},K}{}'$. *Assume that all conditions of Theorem 3.8.4 hold. If* $\zeta_{j,*} \leq \zeta_{j,*}^{+} := r_{\max}\zeta$ *and* $\zeta_{j,\text{new},K} \leq c\zeta$, *then for all* $t \in \tilde{\mathcal{I}}_{j,k}$,

1. $\|\boldsymbol{\Phi}_{(j),K}\boldsymbol{P}_t\|_2 \leq (r_{\max} + c_{\text{new}})\zeta$

2. $\|[(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}]^{-1}\|_2 \leq \phi^{+} := 1.2$.

3. $\hat{\mathcal{T}}_t = \mathcal{T}_t$,

4. $\boldsymbol{e}_t = \boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{\Phi}_{(j),K}\boldsymbol{P}_t\boldsymbol{a}_t$ *and* $\|\boldsymbol{e}_t\|_2 \leq \phi^{+}\sqrt{\zeta}$.

We will use the following fact throughout.

Fact 3.9.3. *As before,*

1. $\Gamma_{j-1,\text{end}}$ *implies* NODETS_{j-1}^{a} *for* $a = u_{j-1}$ *or* $a = u_{j-1} + 1$ *which implies* $\hat{t}_j \geq t_j$.

2. $\text{DET}_j^{u_j}$ *implies that* $\hat{t}_j \leq t_j + \alpha$.

3. $\text{DET}_j^{u_j+1}$ *implies that* $\hat{t}_j \leq t_j + 2\alpha$.

4. *The above facts combined with the model assumption* $d_2 \geq \vartheta\tilde{\alpha} + 2\alpha$ *imply that conditioned on* $\Gamma_{j-1,\text{end}} \cap \text{DET}_j^{u_j+1}$, $\tilde{\mathcal{I}}_{j,k} \subseteq [t_{j+1} - d_2, t_{j+1} - 1]$ *for* $j = 1, \ldots, J$ *and* $k = 1, \ldots, \vartheta$ *and* $\hat{u}_j = u_j$ *or* $\hat{u}_j = u_j + 1$. *Thus, during these intervals, the covariance matrix* $\boldsymbol{\Lambda}_{a,t}$ *is assumed constant and obeys the clustering assumption.*

Remark 3.9.4. *As in the addition proof, we will remove the subscript* j *at various places.*

The following two lemmas are proved the same way as their counterparts in Section 3.6 (Lemmas 3.6.2 and 3.6.3 respectively). As before, when they are applied, we will use $\epsilon = 0.01 c_{\text{new}} \zeta \lambda^-$.

Lemma 3.9.5.

1. $\mathbb{P}\left(\lambda_{\max}\left(\frac{1}{\tilde{\alpha}}\sum_{t\in\tilde{\mathcal{I}}_{j,k}} \boldsymbol{a}_{t,\text{det}}\boldsymbol{a}_{t,\text{det}}{}'\right) \le \frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + (1-b^2)\lambda^+_{j,k-1} + \epsilon \;\middle|\; \tilde{X}_{k-1}\right) \ge 1 - rF(\tilde{\alpha}, \epsilon, r\gamma^2)$
2. $\mathbb{P}\left(\lambda_{\max}\left(\frac{1}{\tilde{\alpha}}\sum_{t\in\tilde{\mathcal{I}}_{j,k}} \boldsymbol{a}_{t,\text{cur}}\boldsymbol{a}_{t,\text{cur}}{}'\right) \le \frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + (1-b^2)\lambda^+_{j,k} + \epsilon \;\middle|\; \tilde{X}_{k-1}\right) \ge 1 - rF(\tilde{\alpha}, \epsilon, r\gamma^2)$
3. $\mathbb{P}\left(\lambda_{\max}\left(\frac{1}{\tilde{\alpha}}\sum_{t\in\tilde{\mathcal{I}}_{j,k}} \boldsymbol{a}_{t,\text{undet}}\boldsymbol{a}_{t,\text{undet}}{}'\right) \le \frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + (1-b^2)\lambda^+_{j,k+1} + \epsilon \;\middle|\; \tilde{X}_{k-1}\right) \ge 1 - rF(\tilde{\alpha}, \epsilon, r\gamma^2)$
4. $\mathbb{P}\left(\lambda_{\min}\left(\frac{1}{\tilde{\alpha}}\sum_{t\in\tilde{\mathcal{I}}_{j,k}} \boldsymbol{a}_{t,\text{cur}}\boldsymbol{a}_{t,\text{cur}}{}'\right) \ge (1-b^2)\lambda^-_{j,k} - \epsilon \;\middle|\; \tilde{X}_{k-1}\right) \ge 1 - rF(\tilde{\alpha}, \epsilon, c_{\text{new}}\gamma^2)$.

Results 1) and 2) also hold for $\boldsymbol{a}_{t,\text{det}}\boldsymbol{a}_t{}'$ and $\boldsymbol{a}_{t,\text{cur}}\boldsymbol{a}_t{}'$ respectively.

Lemma 3.9.6.

1. $$\mathbb{P}\left(\left\|\frac{1}{\tilde{\alpha}}\sum_{t\in\tilde{\mathcal{I}}_{j,k}} \boldsymbol{a}_{t,\text{det}}\boldsymbol{a}_{t,\text{cur}}{}'\right\|_2 \le \frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + 4\epsilon \;\middle|\; \tilde{X}_{k-1}\right) \ge 1 - 3(2r)F(\tilde{\alpha}, \epsilon, 2r\gamma^2) - (2r)F(\tilde{\alpha}\epsilon, 4r\gamma^2).$$

2. $$\mathbb{P}\left(\left\|\frac{1}{\tilde{\alpha}}\sum_{t\in\tilde{\mathcal{I}}_{j,k}} \boldsymbol{a}_{t,\text{det}}\boldsymbol{a}_{t,\text{undet}}{}'\right\|_2 \le \frac{b^2}{\tilde{\alpha}(1-b^2)} r\gamma^2 + 4\epsilon \;\middle|\; \tilde{X}_{k-1}\right) \ge 1 - 3(2r)F(\tilde{\alpha}, \epsilon, 2r\gamma^2) - (2r)F(\tilde{\alpha}\epsilon, 4r\gamma^2).$$

3.

$$\mathbb{P}\left(\left\|\frac{1}{\tilde{\alpha}}\sum_{t\in\tilde{\mathcal{I}}_{j,k}}\boldsymbol{a}_{t,\text{cur}}\boldsymbol{a}_{t,\text{undet}}{}'\right\|_2 \leq \frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2+4\epsilon \;\middle|\; \tilde{X}_{k-1}\right)\geq$$
$$1-3(2r)F(\tilde{\alpha},\epsilon,2r\gamma^2)-(2r)F(\tilde{\alpha}\epsilon,4r\gamma^2).$$

Lemma 3.9.7. *[12] When* $\tilde{X}_{j,k-1}\in\tilde{\Gamma}_{j,k-1}$,

1. $\|\tilde{\boldsymbol{D}}_{j,k,\text{det}}\|_2 \leq r_{\max}\zeta$
2. $\sqrt{1-(r_{\max}+c_{\text{new}})^2\zeta^2} \leq \sigma_i(\tilde{\boldsymbol{R}}_{j,k}) \leq 1$
3. $\|\tilde{\boldsymbol{E}}_{j,k,\text{cur}}{}'\tilde{\boldsymbol{D}}_{j,k,\text{undet}}\|_2 \leq \dfrac{(r_{\max}+c_{\text{new}})^2\zeta^2}{\sqrt{1-(r_{\max}+c_{\text{new}})^2\zeta^2}}$

We are now read to prove Lemmas 3.8.14, 3.8.15, and 3.8.16. Because the lemmas apply for all j, we will often omit the subscript j for convenience. Also, $\sum_t$ will be used to mean $\sum_{t\in\tilde{\mathcal{I}}_{j,k}}$.

Proof of Lemma 3.8.14.

Recall $\tilde{\boldsymbol{A}}_k := \frac{1}{\tilde{\alpha}}\sum_{t\in\tilde{\mathcal{I}}_{j,k}}\tilde{\boldsymbol{E}}_k{}'\left(\boldsymbol{I}-\hat{\boldsymbol{G}}_{\text{det},k}\hat{\boldsymbol{G}}_{\text{det},k}{}'\right)\boldsymbol{\ell}_t\boldsymbol{\ell}_t{}'\left(\boldsymbol{I}-\hat{\boldsymbol{G}}_{\text{det},k}\hat{\boldsymbol{G}}_{\text{det},k}{}'\right)\tilde{\boldsymbol{E}}_k$. Observe that

$$\tilde{\boldsymbol{E}}_k{}'\left(\boldsymbol{I}-\sum_{i=1}^{k-1}\hat{\boldsymbol{G}}_{j,i}\hat{\boldsymbol{G}}_{j,i}{}'\right)\boldsymbol{\ell}_t = \tilde{\boldsymbol{R}}_k\boldsymbol{a}_{t,\text{cur}}+\tilde{\boldsymbol{E}}_k{}'(\tilde{\boldsymbol{D}}_{\text{det},k}\boldsymbol{a}_{t,\text{det}}+\tilde{\boldsymbol{D}}_{\text{undet},k}\boldsymbol{a}_{t,\text{det}})$$

Let $\boldsymbol{Z}_t = \tilde{\boldsymbol{R}}_k\boldsymbol{a}_{t,\text{cur}}$ and $\boldsymbol{Y}_t = \tilde{\boldsymbol{E}}_k{}'(\tilde{\boldsymbol{D}}_{\text{det},k}\boldsymbol{a}_{t,\text{det}}+\tilde{\boldsymbol{D}}_{\text{undet},k}\boldsymbol{a}_{t,\text{undet}})$ Then

$$\tilde{\boldsymbol{A}}_k \succeq \frac{1}{\tilde{\alpha}}\sum_{t\in\tilde{\mathcal{I}}_{j,k}}\boldsymbol{Z}_t\boldsymbol{Z}_t{}'+\boldsymbol{Z}_t\boldsymbol{Y}_t{}'+\boldsymbol{Y}_t\boldsymbol{Z}_t{}' \tag{3.26}$$

Consider $\frac{1}{\tilde{\alpha}}\sum_t\boldsymbol{Z}_t\boldsymbol{Z}_t{}'$. Using a theorem of Ostrowoski [27, Theorem 4.5.9], we have that

$$\lambda_{\min}\left(\frac{1}{\tilde{\alpha}}\sum_t\boldsymbol{Z}_t\boldsymbol{Z}_t{}'\right)\geq\lambda_{\min}(\tilde{\boldsymbol{R}}_k\tilde{\boldsymbol{R}}_k{}')\cdot\lambda_{\min}\left(\frac{1}{\tilde{\alpha}}\sum_t\boldsymbol{a}_{t,\text{cur}}\boldsymbol{a}_{t,\text{cur}}{}'\right)$$

By Lemmas 3.9.5 and 3.9.7, we get that

$$\mathbb{P}\left(\lambda_{\min}\left(\frac{1}{\tilde{\alpha}}\sum_t\boldsymbol{Z}_t\boldsymbol{Z}_t{}'\right)\geq(1-r^2\zeta^2)\left[(1-b^2)\lambda_k^-\epsilon\right]\;\middle|\;\tilde{X}_{k-1}\right)\geq 1-rF(\tilde{\alpha},\epsilon,c_{\text{new}}\gamma^2).$$

for all $\tilde{X}_{k-1}\in\tilde{\Gamma}_{k-1}$.

Next consider the term $\frac{1}{\tilde{\alpha}}\sum_t \boldsymbol{Z}_t\boldsymbol{Y}_t'$.

$$\begin{aligned}\frac{1}{\tilde{\alpha}}\sum_t \boldsymbol{Z}_t\boldsymbol{Y}_t' &= \frac{1}{\tilde{\alpha}}\sum_t \tilde{\boldsymbol{R}}_k\boldsymbol{a}_{t,\text{cur}}\left(\boldsymbol{a}_{t,\text{det}}{}'\tilde{\boldsymbol{D}}_{\text{det},k}{}' + \boldsymbol{a}_{t,\text{undet}}{}'\tilde{\boldsymbol{D}}_{\text{undet},k}{}'\right)\tilde{\boldsymbol{E}}_k \\ &= \frac{1}{\tilde{\alpha}}\sum_t \tilde{\boldsymbol{R}}_k\boldsymbol{a}_{t,\text{cur}}\boldsymbol{a}_{t,\text{det}}{}'\tilde{\boldsymbol{D}}_{\text{det},k}{}'\tilde{\boldsymbol{E}}_k + \tilde{\boldsymbol{R}}_k\boldsymbol{a}_{t,\text{cur}}\boldsymbol{a}_{t,\text{undet}}{}'\tilde{\boldsymbol{D}}_{\text{undet},k}{}'\tilde{\boldsymbol{E}}_k\end{aligned}$$

By Lemmas 3.9.6 and 3.9.7, we get that

$$\mathbb{P}\left(\lambda_{\min}\left(\frac{1}{\tilde{\alpha}}\sum_t \boldsymbol{Z}_t\boldsymbol{Y}_t'\right) \geq -\sqrt{1-r^2\zeta^2}\left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2+4\epsilon\right)\left(r\zeta+\frac{r^2\zeta^2}{\sqrt{1-r^2\zeta^2}}\right) \;\middle|\; \tilde{X}_{k-1}\right) \geq 1-2(3(2r)F(\tilde{\alpha},\epsilon,2r\gamma^2)-(2r)F(\tilde{\alpha}\epsilon,4r\gamma^2)).$$

for all $\tilde{X}_{k-1}\in\tilde{\Gamma}_{k-1}$.

Therefore

$$\begin{aligned}&\mathbb{P}\Bigg(\lambda_{\min}\left(\tilde{\boldsymbol{A}}_{j,k}\right) \geq (1-r^2\zeta^2)\left[(1-b^2)\lambda_k^- - \epsilon\right] \\ &\quad - 2\sqrt{1-r^2\zeta^2}\left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2+4\epsilon\right)\left(r\zeta+\frac{r^2\zeta^2}{\sqrt{1-r^2\zeta^2}}\right) \;\Big|\; \tilde{X}_{k-1}\Bigg) \\ &\quad \geq 1 - rF(\tilde{\alpha},\epsilon,c_{\text{new}}\gamma^2) - 2(3(2r)F(\tilde{\alpha},\epsilon,2r\gamma^2)-(2r)F(\tilde{\alpha}\epsilon,4r\gamma^2)).\end{aligned}$$

for all $\tilde{X}_{k-1}\in\tilde{\Gamma}_{k-1}$.

⊠

Proof of Lemma 3.8.15.

Recall $\tilde{\boldsymbol{A}}_{k,\perp} := \frac{1}{\tilde{\alpha}}\sum_{t\in\tilde{\mathcal{I}}_{j,k}}\tilde{\boldsymbol{E}}_{k,\perp}{}'\left(\boldsymbol{I}-\sum_{i=1}^{k-1}\hat{\boldsymbol{G}}_{j,i}\hat{\boldsymbol{G}}_{j,i}{}'\right)\boldsymbol{\ell}_t\boldsymbol{\ell}_t{}'\left(\boldsymbol{I}-\sum_{i=1}^{k-1}\hat{\boldsymbol{G}}_{j,i}\hat{\boldsymbol{G}}_{j,i}{}'\right)\tilde{\boldsymbol{E}}_{k,\perp}$. Notice that because $\tilde{\boldsymbol{E}}_{k,\perp}{}'\left(\boldsymbol{I}-\sum_{i=1}^{k-1}\hat{\boldsymbol{G}}_{j,i}\hat{\boldsymbol{G}}_{j,i}{}'\right)\boldsymbol{G}_k = \boldsymbol{0}$, $\tilde{\boldsymbol{E}}_{k,\perp}{}'\left(\boldsymbol{I}-\sum_{i=1}^{k-1}\hat{\boldsymbol{G}}_{j,i}\hat{\boldsymbol{G}}_{j,i}{}'\right)\boldsymbol{\ell}_t = \tilde{\boldsymbol{E}}_{k,\perp}{}'(\tilde{\boldsymbol{D}}_{\text{det},k}\boldsymbol{a}_{t,\text{det}}+\tilde{\boldsymbol{D}}_{\text{undet},k}\boldsymbol{a}_{t,\text{undet}})$. So $\tilde{\boldsymbol{A}}_{k,\perp}$ can be written as

$$\tilde{\boldsymbol{A}}_{k,\perp} = \frac{1}{\tilde{\alpha}}\sum_t \tilde{\boldsymbol{E}}_{k,\perp}{}'(\tilde{\boldsymbol{D}}_{\text{det},k}\boldsymbol{a}_{t,\text{det}}+\tilde{\boldsymbol{D}}_{\text{undet},k}\boldsymbol{a}_{t,\text{undet}})(\tilde{\boldsymbol{D}}_{\text{det},k}\boldsymbol{a}_{t,\text{det}}+\tilde{\boldsymbol{D}}_{\text{undet},k}\boldsymbol{a}_{t,\text{undet}})'\tilde{\boldsymbol{E}}_{k,\perp}.$$

Expanding the above gives

$$\begin{aligned}\tilde{\boldsymbol{A}}_{k,\perp} = \frac{1}{\tilde{\alpha}}\sum_t \tilde{\boldsymbol{E}}_{k,\perp}{}'\Big(&\tilde{\boldsymbol{D}}_{\text{det},k}\boldsymbol{a}_{t,\text{det}}\boldsymbol{a}_{t,\text{det}}{}'\tilde{\boldsymbol{D}}_{\text{det},k}{}' + \tilde{\boldsymbol{D}}_{\text{det},k}\boldsymbol{a}_{t,\text{det}}\boldsymbol{a}_{t,\text{undet}}{}'\tilde{\boldsymbol{D}}_{\text{undet},k}{}' + \\ &\tilde{\boldsymbol{D}}_{\text{undet},k}\boldsymbol{a}_{t,\text{undet}}\boldsymbol{a}_{t,\text{det}}{}'\tilde{\boldsymbol{D}}_{\text{det},k}{}' + \tilde{\boldsymbol{D}}_{\text{undet},k}\boldsymbol{a}_{t,\text{undet}}\boldsymbol{a}_{t,\text{undet}}{}'\tilde{\boldsymbol{D}}_{\text{undet},k}{}'\Big)\tilde{\boldsymbol{E}}_{k,\perp}.\end{aligned}$$

By Lemmas 3.9.5, 3.9.6 and 3.9.7,

$$\mathbb{P}\Bigg(\lambda_{\max}\left(\tilde{\boldsymbol{A}}_{j,k,\perp}\right) \leq r^2\zeta^2\left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + (1-b^2)\lambda^+_{j,k-1} + \epsilon\right) + 2r\zeta\left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + 4\epsilon\right) + \left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + (1-b^2)\lambda^+_{j,k+1} + \epsilon\right) \;\Big|\; \tilde{X}_{k-1}\Bigg) \geq 1 - 3r\digamma(\tilde{\alpha}, \epsilon, r\gamma^2)$$

for all $\tilde{X}_{k-1} \in \tilde{\Gamma}_{k-1}$.

⊠

Proof of Lemma 3.8.16. For Ease of Notation, define $\boldsymbol{\Psi}_{j,k} := \left(\boldsymbol{I} - \hat{\boldsymbol{G}}_{j,k,\det}\hat{\boldsymbol{G}}_{j,k,\det}{}'\right)$ and $\tilde{\boldsymbol{F}}_t = \tilde{\boldsymbol{E}}_k\tilde{\boldsymbol{E}}_k{}'\boldsymbol{\Psi}_{k-1}\boldsymbol{\ell}_t\boldsymbol{\ell}_t{}'\boldsymbol{\Psi}_{k-1}\tilde{\boldsymbol{E}}_{k,\perp}\tilde{\boldsymbol{E}}_{k,\perp}{}'$. Using the expression for $\tilde{\boldsymbol{\mathcal{H}}}_{j,k}$ given in Definition 3.8.13, adding and subtracting $\tilde{\boldsymbol{F}}_t + \tilde{\boldsymbol{F}}_t{}'$, and noting that $\tilde{\boldsymbol{E}}_k\tilde{\boldsymbol{E}}_k{}' + \tilde{\boldsymbol{E}}_{k,\perp}\tilde{\boldsymbol{E}}_{k,\perp}{}' = \boldsymbol{I}$ we get that

$$\tilde{\boldsymbol{\mathcal{H}}}_k = \frac{1}{\tilde{\alpha}}\sum_t\Big(\boldsymbol{\Psi}_{k-1}\boldsymbol{e}_t\boldsymbol{e}_t{}'\boldsymbol{\Psi}_{k-1} - (\boldsymbol{\Psi}_{k-1}\boldsymbol{\ell}_t\boldsymbol{e}_t{}'\boldsymbol{\Psi}_{k-1} + \boldsymbol{\Psi}_{k-1}\boldsymbol{e}_t\boldsymbol{\ell}_t{}'\boldsymbol{\Psi}_{k-1}) + (\tilde{\boldsymbol{F}}_t + \tilde{\boldsymbol{F}}_t{}')\Big)$$

Thus

$$\|\tilde{\boldsymbol{\mathcal{H}}}_k\|_2 \leq \left\|\frac{1}{\tilde{\alpha}}\sum_t \boldsymbol{e}_t\boldsymbol{e}_t{}'\right\|_2 + 2\left\|\frac{1}{\tilde{\alpha}}\sum_t \boldsymbol{\Psi}_{k-1}\boldsymbol{\ell}_t\boldsymbol{e}_t{}'\right\|_2 + 2\left\|\frac{1}{\tilde{\alpha}}\sum_t \tilde{\boldsymbol{F}}_t\right\|_2 \tag{3.27}$$

1. First consider $\|\frac{1}{\tilde{\alpha}}\sum_t \boldsymbol{e}_t\boldsymbol{e}_t{}'\|_2$. By Lemma 3.9.2, when $\tilde{X}_{k-1} \in \tilde{\Gamma}_{k-1}$,

$$\boldsymbol{e}_t = \boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{\Phi}_{(j),K}\boldsymbol{P}_t\boldsymbol{a}_t$$

and $\|\boldsymbol{e}_t\|_2 \leq (\phi^+)^2\zeta$. By the same argument as Lemma 3.6.1,

$$\begin{aligned}\mathbb{E}_{t-1}\left[\boldsymbol{e}_t\boldsymbol{e}_t{}'\middle|\tilde{X}_{k-1}\right] &= \boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{\Phi}_{(j),K}\boldsymbol{P}_t\left(b^2\boldsymbol{a}_{t-1}\boldsymbol{a}_{t-1}{}' + \boldsymbol{\Lambda}_{\nu,t}\right)\\ &\qquad \boldsymbol{P}_t{}'\boldsymbol{\Phi}_{(j),K}\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\\ &\preceq \left(\rho^2h^+(\phi^+)^2(r_{\max}+c_{\text{new}})^2\zeta^2\left(b^2r\gamma^2 + (1-b^2)\lambda^+\right)\right)\boldsymbol{I}\end{aligned}$$

Applying the Azuma corollary (Corollary 3.A.9) gives,

$$\mathbb{P}\left(\left\|\frac{1}{\tilde{\alpha}}\sum_t \boldsymbol{e}_t\boldsymbol{e}_t{}'\right\|_2 \leq \rho^2h^+(\phi^+)^2(r_{\max}+c_{\text{new}})^2\zeta^2\left(b^2r\gamma^2 + (1-b^2)\lambda^+\right) + \epsilon \;\middle|\; \tilde{X}_{k-1}\right) \geq 1 - n\digamma(\tilde{\alpha}, \epsilon, (\phi^+)^4\zeta^2) \tag{3.28}$$

for all $\tilde{X}_{k-1} \in \tilde{\Gamma}_{k-1}$.

2. Next consider $\|\frac{1}{\tilde{\alpha}}\sum_t \boldsymbol{\Psi}_{k-1}\boldsymbol{\ell}_t\boldsymbol{e}_t{}'\|_2$.

$$\begin{aligned}
\boldsymbol{\Psi}_{k-1}\boldsymbol{\ell}_t\boldsymbol{e}_t{}' &= \left(\tilde{\boldsymbol{D}}_k\boldsymbol{a}_{t,\text{cur}} + \tilde{\boldsymbol{D}}_{\text{det},k}\boldsymbol{a}_{t,\text{det}} + \tilde{\boldsymbol{D}}_{\text{undet},k}\boldsymbol{a}_{t,\text{undet}}\right)\boldsymbol{e}_t{}' \\
&= \left(\tilde{\boldsymbol{D}}_{\text{det},k}\boldsymbol{a}_{t,\text{det}} + \begin{bmatrix}\tilde{\boldsymbol{D}}_k & \tilde{\boldsymbol{D}}_{\text{undet},k}\end{bmatrix}\begin{bmatrix}\boldsymbol{a}_{t,\text{cur}} \\ \boldsymbol{a}_{t,\text{undet}}\end{bmatrix}\right)\boldsymbol{e}_t{}' \\
&= \left(\tilde{\boldsymbol{D}}_{\text{det},k}\boldsymbol{a}_{t,\text{det}} + \begin{bmatrix}\tilde{\boldsymbol{D}}_k & \tilde{\boldsymbol{D}}_{\text{undet},k}\end{bmatrix}\begin{bmatrix}\boldsymbol{a}_{t,\text{cur}} \\ \boldsymbol{a}_{t,\text{undet}}\end{bmatrix}\right) \\
&\qquad (\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{\Phi}_{(j),K}\boldsymbol{P}_{(j)}\boldsymbol{a}_t)' \\
&= \left(\tilde{\boldsymbol{D}}_{\text{det},k}\boldsymbol{a}_{t,\text{det}}\boldsymbol{a}_t{}' + \begin{bmatrix}\tilde{\boldsymbol{D}}_k & \tilde{\boldsymbol{D}}_{\text{undet},k}\end{bmatrix}\begin{bmatrix}\boldsymbol{a}_{t,\text{cur}} \\ \boldsymbol{a}_{t,\text{undet}}\end{bmatrix}\boldsymbol{a}_t{}'\right) \\
&\qquad (\boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'\boldsymbol{\Phi}_{(j),K}\boldsymbol{P}_{(j)})'
\end{aligned}$$

Let,

$$\boldsymbol{X}_t = \left(\tilde{\boldsymbol{D}}_{\text{det},k}\boldsymbol{a}_{t,\text{det}}\boldsymbol{a}_t{}' + \begin{bmatrix}\tilde{\boldsymbol{D}}_k & \tilde{\boldsymbol{D}}_{\text{undet},k}\end{bmatrix}\begin{bmatrix}\boldsymbol{a}_{t,\text{cur}} \\ \boldsymbol{a}_{t,\text{undet}}\end{bmatrix}\boldsymbol{a}_t{}'\right)\boldsymbol{P}_{(j)}{}'\boldsymbol{\Phi}_{(j),K}$$

and

$$\boldsymbol{Y}_t = \boldsymbol{I}_{\mathcal{T}_t}[(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}{}'(\boldsymbol{\Phi}_{(j),K})_{\mathcal{T}_t}]^{-1}\boldsymbol{I}_{\mathcal{T}_t}{}'$$

Then by Lemma 3.A.3, which is stated and proved in the appendix,

$$\left\|\frac{1}{\tilde{\alpha}}\sum_t \boldsymbol{\Psi}_{k-1}\boldsymbol{\ell}_t\boldsymbol{e}_t{}'\right\|_2^2 \leq \lambda_{\max}\left(\frac{1}{\tilde{\alpha}}\sum_{t\in\tilde{\mathcal{I}}_{j,k}}\boldsymbol{X}_t\boldsymbol{X}_t{}'\right)\lambda_{\max}\left(\frac{1}{\tilde{\alpha}}\sum_{t\in\tilde{\mathcal{I}}_{j,k}}\boldsymbol{Y}_t\boldsymbol{Y}_t{}'\right)$$

The reason for using Lemma 3.A.3 is so that Lemma 3.3.3 can be applied to $\boldsymbol{Y}_t$. For $\boldsymbol{X}_t$, we use Lemmas 3.9.2, 3.9.5, and 3.9.7. This gives,

$$\begin{aligned}
\mathbb{P}\left(\left\|\frac{1}{\tilde{\alpha}}\sum_t \boldsymbol{\Psi}_{k-1}\boldsymbol{\ell}_t\boldsymbol{e}_t{}'\right\|_2 \leq (r\zeta)\left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + (1-b^2)\lambda^+ + \epsilon\right)\right. \\
(r_{\max}+c_{\text{new}})\zeta\left(\sqrt{\rho^2h^+}\phi^+\right) + \\
\left.\left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + (1-b^2)\lambda_{j,k}^+ + \epsilon\right)(r_{\max}+c_{\text{new}})\zeta\left(\sqrt{\rho^2h^+}\phi^+\right) \;\middle|\; \tilde{X}_{k-1}\right) \\
\geq 1 - rF(\tilde{\alpha},\epsilon,r\gamma^2)
\end{aligned}$$

for all $\tilde{X}_{k-1} \in \tilde{\Gamma}_{k-1}$.

3. Finally consider $\|\frac{1}{\tilde{\alpha}}\sum_t \tilde{\boldsymbol{F}}_t\|_2$. Recall that $\tilde{\boldsymbol{F}}_t = \tilde{\boldsymbol{E}}_k\tilde{\boldsymbol{E}}_k{}'\boldsymbol{\Psi}_{k-1}\boldsymbol{\ell}_t\boldsymbol{\ell}_t{}'\boldsymbol{\Psi}_{k-1}\tilde{\boldsymbol{E}}_{k,\perp}\tilde{\boldsymbol{E}}_{k,\perp}{}'$. We will use the fact that $\tilde{\boldsymbol{E}}_{k,\perp}{}'\tilde{\boldsymbol{D}}_k = \boldsymbol{0}$ to simplify this expression.

$$\begin{aligned}\tilde{\boldsymbol{F}}_t &= \tilde{\boldsymbol{E}}_k\tilde{\boldsymbol{E}}_k{}'\boldsymbol{\Psi}_{k-1}\boldsymbol{\ell}_t\boldsymbol{\ell}_t{}'\boldsymbol{\Psi}_{k-1}\tilde{\boldsymbol{E}}_{k,\perp}\tilde{\boldsymbol{E}}_{k,\perp}{}' \\ &= \tilde{\boldsymbol{E}}_k\tilde{\boldsymbol{E}}_k{}'(\tilde{\boldsymbol{D}}_k\boldsymbol{a}_{t,\text{cur}} + \tilde{\boldsymbol{D}}_{\text{det},k}\boldsymbol{a}_{t,\text{det}} + \tilde{\boldsymbol{D}}_{\text{undet},k}\boldsymbol{a}_{t,\text{undet}}) \\ &\quad (\tilde{\boldsymbol{D}}_{\text{det},k}\boldsymbol{a}_{t,\text{det}} + \tilde{\boldsymbol{D}}_{\text{undet},k}\boldsymbol{a}_{t,\text{undet}})'\tilde{\boldsymbol{E}}_{k,\perp}\tilde{\boldsymbol{E}}_{k,\perp}{}'\end{aligned}$$

Therefore,

$$\begin{aligned}\mathbb{P}\Bigg(\left\|\frac{1}{\tilde{\alpha}}\sum_t \tilde{\boldsymbol{F}}_t\right\|_2 \leq & r\zeta\left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + 4\epsilon\right) + \\ & \frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + 4\epsilon + \\ & (r\zeta)^2\left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + (1-b^2)\lambda^+_{j,k-1} + \epsilon\right) + \\ & r\zeta\left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + 4\epsilon\right) + \\ & \frac{(r\zeta)^3}{\sqrt{1-r^2\zeta^2}}\left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + 4\epsilon\right) + \\ & \frac{(r\zeta)^2}{\sqrt{1-r^2\zeta^2}}\left(\frac{b^2}{\tilde{\alpha}(1-b^2)}r\gamma^2 + (1-b^2)\lambda^+_{j,k+1} + \epsilon\right) \;\Big|\; \tilde{X}_{k-1}\Bigg) \geq \\ 1 - 3\Big(3(2r)F(\tilde{\alpha},\epsilon,2r\gamma^2) - (2r)F(\tilde{\alpha}\epsilon,4r\gamma^2)\Big) - 2\Big(rF(\tilde{\alpha},\epsilon,c_{\text{new}}\gamma^2)\Big)\end{aligned}$$

for all $\tilde{X}_{k-1} \in \tilde{\Gamma}_{k-1}$.

Therefore,

$$\begin{aligned}\mathbb{P}\Big(\|\tilde{\boldsymbol{\mathcal{H}}}_{j,k}\|_2 \leq \tilde{b}_2 + 2\tilde{b}_{4,k} + 2\tilde{b}_{6,k} \;\Big|\; \tilde{X}_{k-1}\Big) \geq & 1 - nF(\tilde{\alpha},\epsilon,(\phi^+)^4\zeta^2) \\ & - rF(\tilde{\alpha},\epsilon,r\gamma^2) \\ & - 3\Big(3(2r)F(\tilde{\alpha},\epsilon,2r\gamma^2) - (2r)F(\tilde{\alpha}\epsilon,4r\gamma^2)\Big) - 2\Big(rF(\tilde{\alpha},\epsilon,c_{\text{new}}\gamma^2)\Big).\end{aligned}$$

for all $\tilde{X}_{k-1} \in \tilde{\Gamma}_{k-1}$. The quantities $\tilde{b}_2$, $\tilde{b}_{4,k}$, and $\tilde{b}_{6,k}$ were defined in Definition 3.8.6.

⊠

3.10 Simulations

In this section we show the results of simulation experiments that demonstrate the result we have proven. The results can be seen in Figures 3.10 and 3.11. In both cases, results are averaged over 50 simulations.

For both figures, we set $n = 256$ and $t_{\max} = 8200$. The low-dimensional vectors $\boldsymbol{\ell}_t$ were generated the same way for both figures. There are 3 subspace changes, so $J = 3$. The subspace changed at times $t_1 = 701$, $t_2 = 3701$, and $t_3 = 6201$. The bounds on the entries of $\boldsymbol{a}_t$ and $\boldsymbol{a}_{t,\text{new}}$ were $\gamma = 600$, and $\gamma_{\text{new}} = 5$ respectively. At each subspace change, 4 new directions were added, and 4 were removed. Therefore $c_{j,\text{new}} = c_{j,\text{old}} = 4$ for $j = 1, 2, 3$. The algorithm parameters were set as follows: $\alpha = 100$, $K = 12$, thresh $= \frac{1}{27}$, $\xi = \sqrt{2}$ and $\omega = 0.1$.

The difference in the two figures is how the supports of the sparse vectors $\boldsymbol{x}_t$ are generated. In Figure 3.10, each index of $\boldsymbol{x}_t$ was non-zero with probability 0.0586 (so the size of each support is around 15) independently of other indices and other times t. In Figure 3.11, all of the supports of $\boldsymbol{x}_t$ were of size 10 and obeyed Signal Model 3.2.11 with $\varrho = 2$, and $\beta = 25$ (so $h^+ = \frac{25}{\alpha} = 0.25$).

For this simulated data, we compare ReProCS, ReProCS-cPCA, PCP, and mod-PCP. In order to compare with the batch methods PCP and mod-PCP, we ran those algorithms every 2α frames, using the observations $\boldsymbol{m}_t$ from the previous 2α time instants. As one can see in Figure 3.10, the error made by ReProCS after a subspace change is very similar to that made by PCP. However, as ReProCS recovers the new directions, the error decays exponentially (notice that the y axis is logarithmic) as we have proven in Theorem 3.2.15.

In Figure 3.11, the supports of $\boldsymbol{x}_t$ are highly correlated. This causes a problem for the batch methods and one can see that ReProCS has significantly better recovery compared to PCP and even mod-PCP when the supports of $\boldsymbol{x}_t$ are correlated.

The last thing to notice is the difference between ReProCS and ReProCS-cPCA. Both of the figures demonstrate the results we have proven. That is, for both algorithms the error is initially large, but decays exponentially thereafter. Because ReProCS only adds new directions to its estimate of the subspace, the final error after a subspace change increases with j. The

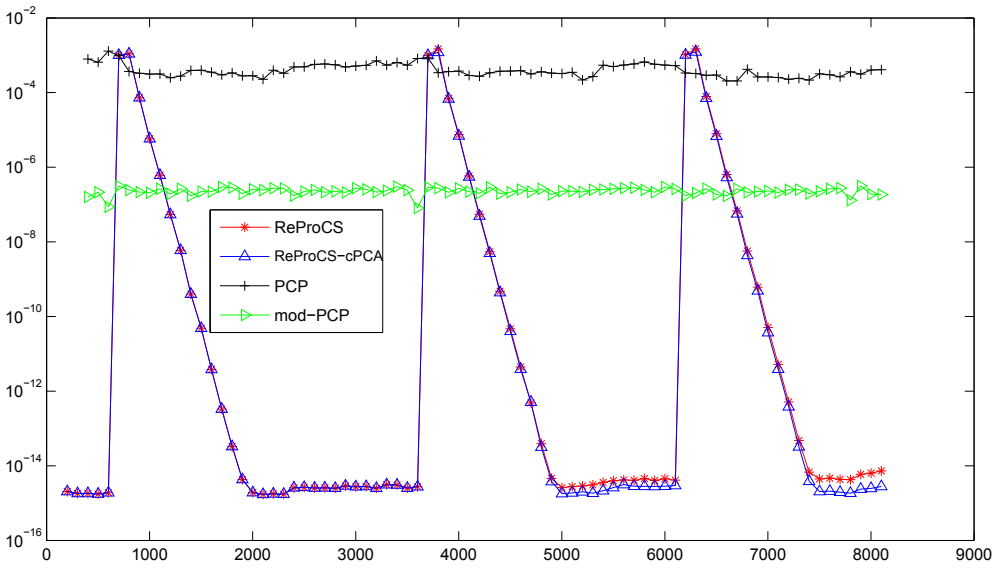

Figure 3.10: Support of $\boldsymbol{X}$ determined by Bernoulli model. The y axis is $\frac{\|\hat{\boldsymbol{x}}_t - \boldsymbol{x}_t\|_2}{\|\boldsymbol{x}_t\|_2}$.

cluster PCA step in ReProCS-cPCA re-estimates the subspace, so that the error decays down to a value that does not increase with j.

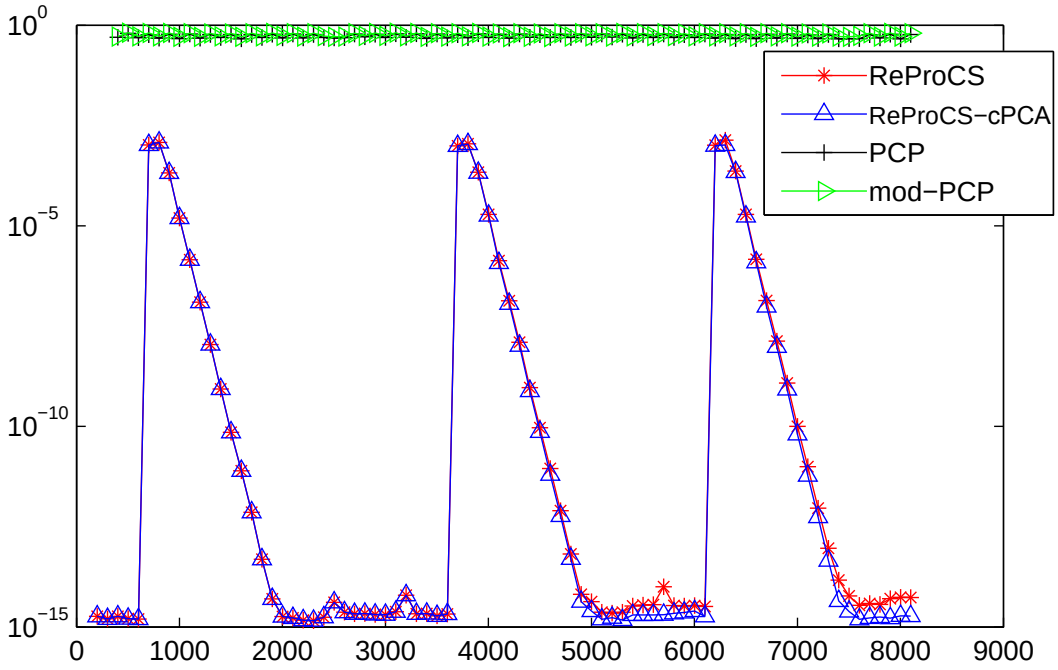

Figure 3.11: Support of $\boldsymbol{X}$ obeys Signal Model 3.2.11. The y axis is $\frac{\|\hat{\boldsymbol{x}}_t - \boldsymbol{x}_t\|_2}{\|\boldsymbol{x}_t\|_2}$.

3.A Preliminaries

Lemma 3.A.1 (Exchanging the order of a double sum)**.**

$$\sum_{t=0}^{\alpha-1}\sum_{i=0}^{t} x_t y_i = \sum_{i=0}^{\alpha-1}\sum_{t=i}^{\alpha-1} x_t y_i$$

Proof. Define [statement] to be the Boolean value of statement. Then,

$$\begin{aligned}
\sum_{t=0}^{\alpha-1}\sum_{i=0}^{t} x_t y_i &= \sum_{t,i} [0 \le i \le t][0 \le t \le \alpha - 1] x_t y_i \\
&= \sum_{t,i} [0 \le i \le t \le \alpha - 1] x_t y_i \\
&= \sum_{t,i} [0 \le i \le \alpha - 1][i \le t \le \alpha - 1] x_t y_i \\
&= \sum_{i=0}^{\alpha-1}\sum_{t=i}^{\alpha-1} x_t y_i
\end{aligned}$$

⊠

Lemma 3.A.2 (Cauchy-Schwarz for a sum of vectors)**.** *For vectors $\boldsymbol{x}_t$ and $\boldsymbol{y}_t$,*

$$\left(\sum_{t=1}^{\alpha} \boldsymbol{x}_t{}'\boldsymbol{y}_t\right)^2 \leq \left(\sum_t \|\boldsymbol{x}_t\|_2^2\right)\left(\sum_t \|\boldsymbol{y}_t\|_2^2\right)$$

Proof.

$$\left(\sum_{t=1}^{\alpha} \boldsymbol{x}_t{}'\boldsymbol{y}_t\right)^2 = \left([\boldsymbol{x}_1{}', \ldots, \boldsymbol{x}_\alpha{}'] \begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_\alpha \end{bmatrix}\right)^2 \leq \left\| \begin{bmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_\alpha \end{bmatrix} \right\|_2^2 \left\| \begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_\alpha \end{bmatrix} \right\|_2^2 = \left(\sum_{t=1}^{\alpha} \|\boldsymbol{x}_t\|_2^2\right)\left(\sum_{t=1}^{\alpha} \|\boldsymbol{y}_t\|_2^2\right)$$

The inequality is by Cauchy-Schwarz for a single vector. ⊠

Lemma 3.A.3 (Cauchy-Schwarz for a sum of matrices)**.** *For matrices $\boldsymbol{X}_t$ and $\boldsymbol{Y}_t$,*

$$\left\|\frac{1}{\alpha}\sum_{t=1}^{\alpha} \boldsymbol{X}_t\boldsymbol{Y}_t{}'\right\|_2^2 \leq \lambda_{\max}\left(\frac{1}{\alpha}\sum_{t=1}^{\alpha} \boldsymbol{X}_t\boldsymbol{X}_t{}'\right)\lambda_{\max}\left(\frac{1}{\alpha}\sum_{t=1}^{\alpha} \boldsymbol{Y}_t\boldsymbol{Y}_t{}'\right)$$

Proof.

$$\begin{aligned}
\left\|\sum_{t=1}^{\alpha} \boldsymbol{X}_t\boldsymbol{Y}_t{}'\right\|_2^2 &= \max_{\substack{\|\boldsymbol{x}\|=1 \\ \|\boldsymbol{y}\|=1}} \left|\boldsymbol{x}'\left(\sum_t \boldsymbol{X}_t\boldsymbol{Y}_t{}'\right)\boldsymbol{y}\right|^2 \\
&= \max_{\substack{\|\boldsymbol{x}\|=1 \\ \|\boldsymbol{y}\|=1}} \left|\sum_{t=1}^{\alpha} (\boldsymbol{X}_t{}'\boldsymbol{x})'(\boldsymbol{Y}_t{}'\boldsymbol{y})\right|^2 \\
&\leq \max_{\substack{\|\boldsymbol{x}\|=1 \\ \|\boldsymbol{y}\|=1}} \left(\sum_{t=1}^{\alpha} \left\|\boldsymbol{X}_t{}'\boldsymbol{x}\right\|_2^2\right)\left(\sum_{t=1}^{\alpha} \left\|\boldsymbol{Y}_t{}'\boldsymbol{y}\right\|_2^2\right) \\
&= \max_{\|\boldsymbol{x}\|=1} \boldsymbol{x}'\sum_{t=1}^{\alpha} \boldsymbol{X}_t\boldsymbol{X}_t{}'\,\boldsymbol{x} \cdot \max_{\|\boldsymbol{y}\|=1} \boldsymbol{y}'\sum_{t=1}^{\alpha} \boldsymbol{Y}_t\boldsymbol{Y}_t{}'\,\boldsymbol{y} \\
&= \lambda_{\max}\left(\sum_{t=1}^{\alpha} \boldsymbol{X}_t\boldsymbol{X}_t{}'\right)\lambda_{\max}\left(\sum_{t=1}^{\alpha} \boldsymbol{Y}_t\boldsymbol{Y}_t{}'\right)
\end{aligned}$$

The inequality is by Lemma 3.A.2. The penultimate line is because $\|\boldsymbol{v}\|_2^2 = \boldsymbol{v}'\boldsymbol{v}$ (used with $\boldsymbol{v} = \boldsymbol{X}_t{}'\boldsymbol{x}$). Multiplying both sides by $\left(\frac{1}{\alpha}\right)^2$ gives the desired result. ⊠

Lemma 3.A.4. *Let X, Y, and Z be random variables. Assume that X is independent of $\{Y, Z\}$. Then*

$$\mathbb{E}[XY|Z] = \mathbb{E}[X]\mathbb{E}[Y|Z]$$

Proof. By the chain rule, $f_{X,Y|Z}(x,y|z) = f_{X|Y,Z}(x|y,z)f_{Y|Z}(y|z)$. Because X is independent of both Y and Z, $f_{X|Y,Z}(x|y,z) = f_X(x)$. ⊠

Remark 3.A.5. *Adopt the notation that if the lower limit of a sum exceeds the upper limit, then the sum is empty and therefore equal to zero.*

Lemma 3.A.6. *Let $\boldsymbol{c}_\tau$ be a sequence of vectors such that*

$$\boldsymbol{c}_\tau = b\boldsymbol{c}_{\tau-1} + \boldsymbol{\mu}_\tau$$

for a scalar b. Similarly, let $\tilde{\boldsymbol{c}}_\tau = b\tilde{\boldsymbol{c}}_{\tau-1} + \tilde{\boldsymbol{\mu}}_\tau$. Then

$$\sum_{\tau=0}^{\alpha-1} \boldsymbol{c}_\tau \tilde{\boldsymbol{c}}_\tau{}' = \sum_{i=0}^{\alpha-1} [\boldsymbol{Z}_{1,i} + \boldsymbol{Z}_{2,i} + \boldsymbol{Z}_{3,i} + \boldsymbol{Z}_{4,i}] + \boldsymbol{Z}_5$$

where

$$\begin{aligned}
\boldsymbol{Z}_{1,i} &= \frac{(1-b^{2(\alpha-i)})}{1-b^2}\boldsymbol{\mu}_i\tilde{\boldsymbol{\mu}}_i{}', \\
\boldsymbol{Z}_{2,i} &= \sum_{i_2=0}^{i-1} \frac{(1-b^{2(\alpha-i)})}{1-b^2} b^{i-i_2}\boldsymbol{\mu}_i\tilde{\boldsymbol{\mu}}_{i_2}{}', \\
\boldsymbol{Z}_{3,i} &= \sum_{i_2=\alpha-i}^{\alpha-1} \frac{(1-b^{2(\alpha-i_2)})}{1-b^2} b^{i+i_2-\alpha+1}\boldsymbol{\mu}_{\alpha-i-1}\tilde{\boldsymbol{\mu}}_{i_2}{}', \\
\boldsymbol{Z}_{4,i} &= \frac{b^{i+1}(1-b^{2(\alpha-i)})}{1-b^2}(\boldsymbol{\mu}_i\tilde{\boldsymbol{c}}_{-1}{}' + \boldsymbol{c}_{-1}\tilde{\boldsymbol{\mu}}_i{}') \\
\boldsymbol{Z}_5 &= \frac{b^2(1-b^{2\alpha})}{1-b^2}\boldsymbol{c}_{-1}\tilde{\boldsymbol{c}}_{-1}{}'
\end{aligned}$$

Proof. We start with some simple expansions.

$$\begin{aligned}
\sum_{\tau=0}^{\alpha-1} \boldsymbol{c}_\tau \tilde{\boldsymbol{c}}_\tau{}' &= \sum_{\tau=0}^{\alpha-1} (b\boldsymbol{c}_{\tau-1} + \boldsymbol{\mu}_\tau)(b\tilde{\boldsymbol{c}}_{\tau-1} + \tilde{\boldsymbol{\mu}}_\tau)' \\
&= \sum_{\tau=0}^{\alpha-1} \left(b^{\tau+1}\boldsymbol{c}_{-1} + \sum_{i=0}^{\tau} b^{\tau-i}\boldsymbol{\mu}_i \right) \left(b^{\tau+1}\tilde{\boldsymbol{c}}_{-1} + \sum_{i=0}^{\tau} b^{\tau-i}\tilde{\boldsymbol{\mu}}_i \right)' \\
&= \sum_{\tau=0}^{\alpha-1} b^{2(\tau+1)}\boldsymbol{c}_{-1}\tilde{\boldsymbol{c}}_{-1}{}' + \sum_{\tau=0}^{\alpha-1}\sum_{i=0}^{\tau}\sum_{i_2=0}^{\tau} b^{2\tau-i-i_2}\boldsymbol{\mu}_i\tilde{\boldsymbol{\mu}}_{i_2}{}' + \\
&\qquad \sum_{\tau=0}^{\alpha-1}\sum_{i=0}^{\tau} b^{2\tau+1-i}\boldsymbol{c}_{-1}\tilde{\boldsymbol{\mu}}_i{}' + b^{2\tau+1-i}\boldsymbol{\mu}_i\tilde{\boldsymbol{c}}_{-1}{}' \qquad (3.29)
\end{aligned}$$

By summing over the b's, the first term is equal to $\boldsymbol{Z}_5$. Applying Lemma 3.A.1 to the cross terms and summing over the b's, we get

$$\begin{aligned}\sum_{\tau=0}^{\alpha-1}\sum_{i=0}^{\tau} b^{2\tau+1-i}\boldsymbol{c}_{-1}\tilde{\boldsymbol{\mu}}_i{}' + b^{2\tau+1-i}\boldsymbol{\mu}_i\tilde{\boldsymbol{c}}_{-1}{}' &= \sum_{i=0}^{\alpha-1}\sum_{\tau=i}^{\alpha-1} b^{2\tau+1-i}\left(\boldsymbol{c}_{-1}\tilde{\boldsymbol{\mu}}_i{}' + \boldsymbol{\mu}_i\tilde{\boldsymbol{c}}_{-1}{}'\right)\\ &= \sum_{i=0}^{\alpha-1}\frac{b^{i+1}(1-b^{2(\alpha-i)})}{1-b^2}(\boldsymbol{\mu}_i\tilde{\boldsymbol{c}}_{-1}{}' + \boldsymbol{c}_{-1}\tilde{\boldsymbol{\mu}}_i{}')\\ &= \sum_{i=0}^{\alpha-1}\boldsymbol{Z}_{4,i}\end{aligned}$$

The remaining three terms, $\boldsymbol{Z}_{1,i}$, $\boldsymbol{Z}_{2,i}$, and $\boldsymbol{Z}_{3,i}$ will all come from the middle term in (3.29) We apply Lemma 3.A.1 to term2, split the innermost sum into 3 parts, and use the fact that $\sum_{i=0}^{\alpha-1} x_i = \sum_{i=0}^{\alpha-1} x_{\alpha-1-i}$

$$\begin{aligned}&\sum_{\tau=0}^{\alpha-1}\sum_{i=0}^{\tau}\sum_{i_2=0}^{\tau} b^{2\tau-i-i_2}\boldsymbol{\mu}_i\tilde{\boldsymbol{\mu}}_{i_2}{}' && = \sum_{i=0}^{\alpha-1}\sum_{\tau=i}^{\alpha-1}\sum_{i_2=0}^{\tau} b^{2\tau-i-i_2}\boldsymbol{\mu}_i\tilde{\boldsymbol{\mu}}_{i_2}{}'\\ &\text{(split inner sum)} && = \sum_{i=0}^{\alpha-1}\sum_{\tau=i}^{\alpha-1}\sum_{i_2=0}^{i-1} b^{2\tau-i-i_2}\boldsymbol{\mu}_i\tilde{\boldsymbol{\mu}}_{i_2}{}' + \sum_{i=0}^{\alpha-1}\sum_{\tau=i}^{\alpha-1} b^{2\tau-2i}\boldsymbol{\mu}_i\tilde{\boldsymbol{\mu}}_i{}' +\\ &&& \sum_{i=0}^{\alpha-1}\sum_{\tau=i}^{\alpha-1}\sum_{i_2=i+1}^{\tau} b^{2\tau-i-i_2}\boldsymbol{\mu}_i\tilde{\boldsymbol{\mu}}_{i_2}{}'\\ &\text{(switch indices)} && = \sum_{i=0}^{\alpha-1}\sum_{i_2=0}^{i-1}\sum_{\tau=i}^{\alpha-1} b^{2\tau-i-i_2}\boldsymbol{\mu}_i\tilde{\boldsymbol{\mu}}_{i_2}{}' + \sum_{i=0}^{\alpha-1}\sum_{\tau=i}^{\alpha-1} b^{2\tau-2i}\boldsymbol{\mu}_i\tilde{\boldsymbol{\mu}}_i{}' +\\ &&& \sum_{i=0}^{\alpha-1}\sum_{i_2=i+1}^{\alpha-1}\sum_{\tau=i_2}^{\alpha-1} b^{2\tau-i-i_2}\boldsymbol{\mu}_i\tilde{\boldsymbol{\mu}}_{i_2}{}'\\ &\text{(apply fact)} && = \sum_{i=0}^{\alpha-1}\sum_{i_2=0}^{i-1}\sum_{\tau=i}^{\alpha-1} b^{2\tau-i-i_2}\boldsymbol{\mu}_i\tilde{\boldsymbol{\mu}}_{i_2}{}' + \sum_{i=0}^{\alpha-1}\sum_{\tau=i}^{\alpha-1} b^{2\tau-2i}\boldsymbol{\mu}_i\tilde{\boldsymbol{\mu}}_i{}'\\ &&& + \sum_{i=0}^{\alpha-1}\sum_{i_2=\alpha-i}^{\alpha-1}\sum_{\tau=i_2}^{\alpha-1} b^{2\tau-(\alpha-1-i)-i_2}\boldsymbol{\mu}_{\alpha-1-i}\tilde{\boldsymbol{\mu}}_{i_2}{}'\\ &\text{(geometric series)} && = \sum_{i=0}^{\alpha-1}[\boldsymbol{Z}_{2,i} + \boldsymbol{Z}_{1,i} + \boldsymbol{Z}_{3,i}]\end{aligned}$$

⊠

Fact 3.A.7. *For an event $\mathcal{E}$ and random variable X, $\mathbb{P}(\mathcal{E}|X) \geq p$ for all $X \in \mathcal{C}$ implies that $\mathbb{P}(\mathcal{E}|X \in \mathcal{C}) \geq p$.*

Theorem 3.A.8 (Matrix Azuma). *[11, Theorem 7.1] Consider a finite adapted sequence $\boldsymbol{Z}_t$ of $n \times n$ random Hermitian matrices, and a fixed sequence $\boldsymbol{A}_t$ of Hermitian matrices that satisfy*

$$\mathbb{E}_{t-1}[\boldsymbol{Z}_t] = \mathbf{0} \quad \textit{and} \quad {\boldsymbol{Z}_t}^2 \preceq {\boldsymbol{A}_t}^2 \quad \textit{with probability 1.}$$

Define the variance parameter

$$\sigma^2 := \left\| \sum_t {\boldsymbol{A}_t}^2 \right\|_2.$$

Then, for all $\epsilon > 0$,

$$\mathbb{P}\left(\lambda_{\max}\left(\sum_t \boldsymbol{Z}_t \right) \geq \epsilon \right) \leq n \exp\left(\frac{-\epsilon^2}{8\sigma^2} \right)$$

The following corollary extends the above result to the case where $\mathbb{E}_{t-1}[\boldsymbol{Z}_t] \neq \mathbf{0}$ and also includes conditioning on another random variable.

Corollary 3.A.9 (Matrix Azuma conditioned on another random variable for a nonzero mean Hermitian matrix). *Consider an α-length sequence $\{\boldsymbol{Z}_t\}_{1 \leq t \leq \alpha}$ of random Hermitian matrices of size $n \times n$ given a random variable X. Assume that, for all $X \in \mathcal{C}$, (i) $\mathbb{P}(b_1 \boldsymbol{I} \preceq Z_t \preceq b_2 \boldsymbol{I} | X) = 1$, for $1 \leq t \leq \alpha$ and (ii) $b_3 \boldsymbol{I} \preceq \frac{1}{\alpha} \sum_{t=1}^{\alpha} \mathbb{E}_{t-1}(Z_t | X) \preceq b_4 \boldsymbol{I}$. Then for all $\epsilon > 0$,*

$$\mathbb{P}\left(\lambda_{\max}\left(\frac{1}{\alpha} \sum_{t=1}^{\alpha} \boldsymbol{Z}_t \right) \leq b_4 + \epsilon \middle| X \right) \geq 1 - n \exp\left(\frac{-\alpha\epsilon^2}{8(b_2 - b_1)^2} \right)$$

$$\mathbb{P}\left(\lambda_{\min}\left(\frac{1}{\alpha} \sum_{t=1}^{\alpha} \boldsymbol{Z}_t \right) \geq b_3 - \epsilon \middle| X \right) \geq 1 - n \exp\left(\frac{-\alpha\epsilon^2}{8(b_2 - b_1)^2} \right)$$

Proof. 1. Let $\boldsymbol{Y}_t := \boldsymbol{Z}_t - \mathbb{E}_{t-1}(\boldsymbol{Z}_t | X)$. Clearly $\mathbb{E}_{t-1}(\boldsymbol{Y}_t | X) = \mathbf{0}$. Since for all $X \in \mathcal{C}$, $\mathbb{P}(b_1 \boldsymbol{I} \preceq \boldsymbol{Z}_t \preceq b_2 \boldsymbol{I} | X) = 1$ and since for an Hermitian matrix, $\lambda_{\max}(.)$ is a convex function, and $\lambda_{\min}(.)$ is a concave function, $b_1 \boldsymbol{I} \preceq \mathbb{E}_{t-1}(Z_t | X) \preceq b_2 \boldsymbol{I}$ for all $X \in \mathcal{C}$. Therefore, $\mathbb{P}({\boldsymbol{Y}_t}^2 \preceq (b_2 - b_1)^2 \boldsymbol{I} | X) = 1$ for all $X \in \mathcal{C}$. Thus, for Theorem 3.A.8, $\sigma^2 = \| \sum_{t=1}^{\alpha} (b_2 - b_1)^2 \boldsymbol{I} \|_2 = \alpha (b_2 - b_1)^2$. For any $X \in \mathcal{C}$, applying Theorem 3.A.8 for $\{\boldsymbol{Y}_t\}_{t=1,\ldots,\alpha}$ conditioned on X, we get that, for any $\epsilon > 0$,

$$\mathbb{P}\left(\lambda_{\max}\left(\frac{1}{\alpha} \sum_{t=1}^{\alpha} \boldsymbol{Y}_t \right) \leq \epsilon \middle| X \right) > 1 - n \exp\left(\frac{-\alpha\epsilon^2}{8(b_2 - b_1)^2} \right) \text{ for all } X \in \mathcal{C}$$

By Weyl's theorem, $\lambda_{\max}(\frac{1}{\alpha} \sum_{t=1}^{\alpha} \boldsymbol{Y}_t) = \lambda_{\max}(\frac{1}{\alpha} \sum_{t=1}^{\alpha} (\boldsymbol{Z}_t - \mathbb{E}_{t-1}(\boldsymbol{Z}_t | X)) \geq \lambda_{\max}(\frac{1}{\alpha} \sum_{t=1}^{\alpha} \boldsymbol{Z}_t) + \lambda_{\min}(\frac{1}{\alpha} \sum_{t=1}^{\alpha} -\mathbb{E}_{t-1}(\boldsymbol{Z}_t | X))$.

Since $\lambda_{\min}(\frac{1}{\alpha}\sum_{t=1}^{\alpha} -\mathbb{E}_{t-1}(\boldsymbol{Z}_t|X)) = -\lambda_{\max}(\frac{1}{\alpha}\sum_{t=1}^{\alpha}\mathbb{E}_{t-1}(\boldsymbol{Z}_t|X)) \geq -b_4$, thus $\lambda_{\max}(\frac{1}{\alpha}\sum_{t=1}^{\alpha}\boldsymbol{Y}_t) \geq \lambda_{\max}(\frac{1}{\alpha}\sum_{t=1}^{\alpha}\boldsymbol{Z}_t) - b_4$. Therefore,

$$\mathbb{P}\left(\lambda_{\max}\left(\frac{1}{\alpha}\sum_{t=1}^{\alpha}\boldsymbol{Z}_t\right) \leq b_4 + \epsilon \middle| X\right) > 1 - n\exp\left(\frac{-\alpha\epsilon^2}{8(b_2-b_1)^2}\right) \text{ for all } X \in \mathcal{C}$$

2. Now let $\boldsymbol{Y}_t = \mathbb{E}_{t-1}(\boldsymbol{Z}_t|X) - \boldsymbol{Z}_t$. As before, $\mathbb{E}_{t-1}(\boldsymbol{Y}_t|X) = 0$ and conditioned on any $X \in \mathcal{C}$, $\mathbf{P}(Y_t^2 \preceq (b_2-b_1)^2\boldsymbol{I}|X) = 1$. As before, applying Theorem 3.A.8, we get that for any $\epsilon > 0$,

$$\mathbb{P}\left(\lambda_{\max}\left(\frac{1}{\alpha}\sum_{t=1}^{\alpha}\boldsymbol{Y}_t\right) \leq \epsilon \middle| X\right) > 1 - n\exp\left(\frac{-\alpha\epsilon^2}{8(b_2-b_1)^2}\right) \text{ for all } X \in \mathcal{C}$$

By Weyl's theorem, $\lambda_{\max}(\frac{1}{\alpha}\sum_{t=1}^{\alpha}\boldsymbol{Y}_t) = \lambda_{\max}(\frac{1}{\alpha}\sum_{t=1}^{\alpha}(\mathbb{E}_{t-1}(\boldsymbol{Z}_t|X) - \boldsymbol{Z}_t)) \geq \lambda_{\min}(\frac{1}{\alpha}\sum_{t=1}^{\alpha}\mathbb{E}_{t-1}(\boldsymbol{Z}_t|X)) + \lambda_{\max}(\frac{1}{\alpha}\sum_{t=1}^{\alpha}-\boldsymbol{Z}_t) = \lambda_{\min}(\frac{1}{\alpha}\sum_{t=1}^{\alpha}\mathbb{E}_{t-1}(\boldsymbol{Z}_t|X)) - \lambda_{\min}(\frac{1}{\alpha}\sum_{t=1}^{\alpha}\boldsymbol{Z}_t) \geq b_3 - \lambda_{\min}(\frac{1}{\alpha}\sum_{t=1}^{\alpha}\boldsymbol{Z}_t)$ Therefore, for any $\epsilon > 0$,

$$\mathbb{P}\left(\lambda_{\min}\left(\frac{1}{\alpha}\sum_{t=1}^{\alpha}\boldsymbol{Z}_t\right) \geq b_3 - \epsilon \middle| X\right) \geq 1 - n\exp\left(\frac{-\alpha\epsilon^2}{8(b_2-b_1)^2}\right) \text{ for all } X \in \mathcal{C}$$

⊠

We can further extend this to the case of a matrix which is not necessarily Hermitian.

Corollary 3.A.10 (Matrix Azuma conditioned on another random variable for an arbitrary matrix)**.** *Consider an α-length adapted sequence $\{\boldsymbol{Z}_t\}$ of random matrices of size $n_1 \times n_2$ given a random variable X. Assume that, for all $X \in \mathcal{C}$, (i) $\mathbb{P}(\|\boldsymbol{Z}_t\|_2 \leq b_1|X) = 1$ and (ii) $\|\frac{1}{\alpha}\sum_{t=1}^{\alpha}\mathbb{E}_{t-1}(\boldsymbol{Z}_t|X)\|_2 \leq b_2$. Then, for all $\epsilon > 0$,*

$$\mathbb{P}\left(\left\|\frac{1}{\alpha}\sum_{t=1}^{\alpha}\boldsymbol{Z}_t\right\|_2 \leq b_2 + \epsilon \middle| X\right) \geq 1 - (n_1+n_2)\exp\left(\frac{-\alpha\epsilon^2}{8(2b_1)^2}\right)$$

Proof. Define the dilation of an $n_1 \times n_2$ matrix $\boldsymbol{M}$ as $\text{dilation}(\boldsymbol{M}) := \begin{bmatrix} \mathbf{0} & \boldsymbol{M}' \\ \boldsymbol{M} & \mathbf{0} \end{bmatrix}$. Notice that this is an $(n_1+n_2)\times(n_1+n_2)$ Hermitian matrix [11] . As shown in [11, equation 2.12],

$$\lambda_{\max}\big(\text{dilation}(\boldsymbol{M})\big) = \|\,\text{dilation}(\boldsymbol{M})\|_2 = \|\boldsymbol{M}\|_2 \tag{3.30}$$

Thus, the corollary assumptions imply that $\mathbf{P}(\|\operatorname{dilation}(\boldsymbol{Z}_t)\|_2 \leq b_1 | X) = 1$ for all $X \in \mathcal{C}$. By (3.30) and the definition of dilation,

$$\frac{1}{\alpha}\sum_t \mathbb{E}_{t-1}[\operatorname{dilation}(\boldsymbol{Z}_t)|X] = \operatorname{dilation}\left(\frac{1}{\alpha}\sum_t \mathbb{E}_{t-1}[\boldsymbol{Z}_t|X]\right) \preceq b_2 \boldsymbol{I}$$

Thus, applying Corollary 3.A.9 to the sequence $\{\operatorname{dilation}(\boldsymbol{Z}_t)\}_{t=1,\ldots,\alpha}$, we get that,

$$\mathbb{P}\left(\lambda_{\max}\left(\frac{1}{\alpha}\sum_{t=1}^{\alpha}\operatorname{dilation}(\boldsymbol{Z}_t)\right) \leq b_2 + \epsilon \middle| X\right) \geq 1 - (n_1+n_2)\exp\left(\frac{-\alpha\epsilon^2}{32 b_1^2}\right) \text{ for all } X \in \mathcal{C}$$

Using (3.30), $\lambda_{\max}\big(\frac{1}{\alpha}\sum_{t=1}^{\alpha}\operatorname{dilation}(\boldsymbol{Z}_t)\big) = \lambda_{\max}\big(\operatorname{dilation}(\frac{1}{\alpha}\sum_{t=1}^{\alpha}\boldsymbol{Z}_t)\big) = \|\frac{1}{\alpha}\sum_{t=1}^{\alpha}\boldsymbol{Z}_t\|_2$ gives the final result. ⊠

3.B Proofs of Support Change Lemmas

Proof of Lemma 3.2.18. We now provide the complete proof for 1) and 2) in the proof sketch.

For 1) we have by the bound in [24]

$$\begin{aligned}
&\mathbb{P}\left(\text{The object moves at least once every } \beta \text{ instants in the interval } \mathcal{J}_u\right) \\
&= \mathbb{P}\left(\text{The bit sequence } \theta_{(u-1)\alpha+1} \dots \theta_{u\alpha} \text{ does not contain a sequence of } \beta \text{ consecutive zeros}\right) \\
&\geq \left(1-(1-q)^{\beta}\right)^{\alpha-\beta+1} \\
&\geq \left(1-(1-q)^{\beta}\right)^{\alpha}.
\end{aligned}$$

We need the object to move at least once every β time instants in *every* interval $\mathcal{J}_u$. We have

$$\begin{aligned}
\mathbb{P}\left(\text{The object moves at least once every } \beta \text{ instants in } \textit{every } \mathcal{J}_u\right) &\geq \left(1-(1-q)^{\beta}\right)^{\lceil \frac{t_{\max}}{\alpha} \rceil \alpha} \\
&\geq \left(1-(1-q)^{\beta}\right)^{(\frac{t_{\max}}{\alpha}+1)\alpha} \\
&\geq 1-(t_{\max}+\alpha)(1-q)^{\beta}.
\end{aligned}$$

This probability will be greater than $1-\frac{n^{-10}}{2}$ if

$$q \geq 1-\left(\frac{n^{-10}}{2(t_{\max}+\alpha)}\right)^{\frac{1}{\beta}}$$

To prove 2), consider the probability of having motion of at least $\frac{s}{\varrho}$ indices whenever the object moves. This will happen if $\mu_t \geq -0.1\frac{s}{\varrho}$ for $t=1,\dots,t_{\max}$. Also, if $\mu_t \leq .1\frac{s}{\varrho}$, then the

object will move by fewer than $1.2\frac{s}{\varrho}$ indices. Using a standard Gaussian tail bound,

$$\begin{aligned}
\mathbb{P}\left(|\mu_t| \leq \frac{0.1s}{\varrho}\right)^{t_{\max}} &\geq \left(1 - \frac{2\exp\left(\frac{-\left(\frac{0.1s}{\varrho}\right)^2}{2\sigma^2}\right)}{\frac{0.1s}{\varrho\sigma}\sqrt{2\pi}}\right)^{t_{\max}} \\
&= \left(1 - \frac{20\sigma\varrho\exp\left(\frac{-(0.1s)^2}{2\varrho^2\sigma^2}\right)}{s\sqrt{2\pi}}\right)^{t_{\max}} \\
&\geq \left(1 - \frac{20\sqrt{\frac{1}{4000\log(n)}}\exp\left(-20\log(n)\right)}{\sqrt{2\pi}}\right)^{t_{\max}} \\
&\geq 1 - t_{\max}\frac{20}{\sqrt{2\pi}}\sqrt{\frac{1}{4000\log(n)}}\exp\left(-20\log(n)\right) \\
&= 1 - t_{\max}\frac{1}{\sqrt{20\pi}}\frac{n^{-20}}{\sqrt{\log(n)}} \\
&\geq 1 - \frac{1}{\sqrt{20\pi}}\frac{n^{10-20}}{\sqrt{\log(n)}} \\
&\geq 1 - \frac{n^{-10}}{2}.
\end{aligned}$$

For simplicity we assume $n \geq 2$, so that $\sqrt{20\pi\log(n)} \geq 2$.

Finally, by the union bound

$$\mathbb{P}\Big(2) \text{ and } 3) \text{ hold}\Big) = 1 - \mathbb{P}\Big(2) \text{ or } 3) \text{ does not hold}\Big) \geq 1 - 2\frac{n^{-10}}{2} = 1 - n^{-10}$$

⊠

Proof of Lemma 3.3.4. Consider an interval of length α. The assumption that the maximum motion each time is by at at most $\varrho_2 s$ indices with $\varrho_2 s\alpha \leq n$ ensures that once an index is removed from the support, it does not return for the next α time instants. Thus without loss of generality, we can renumber the indices so that the object starts at index 1 at the beginning of the interval.

Notice from the model that for a given choice of $\mathcal{T}_{(i)}$'s, $h(\alpha)$ is an upper bound on $h^*(\alpha)$. Thus, as long as we can construct one set of mutually disjoint $\mathcal{T}_{(i)}$'s for which (3.6) holds and for which $h(\alpha; \{\mathcal{T}_{(i),u}\}) \leq h^+\alpha$ we will be done.

Let $\mathcal{T}^{[j]}$ for $j = 1, \ldots, m$ be the distinct supports (in order) of $\boldsymbol{x}_t$ for $t \in \mathcal{J}_u$. That is $\mathcal{T}_t = \mathcal{T}^{[j]}$ for some j. Now define $\mathcal{T}_{(i)} = \mathcal{T}^{[i]} \setminus \mathcal{T}^{[i+1]}$ for $i = 1, \ldots, m-1$, and $\mathcal{T}_{(m)} = \mathcal{T}^{[m]}$. We will show that these $\mathcal{T}_{(i)}$ are disjoint. For a $j > i$,

$$\begin{aligned}\mathcal{T}_{(i)} \cap \mathcal{T}_{(j)} &= \left(\mathcal{T}^{[i]} \setminus \mathcal{T}^{[i+1]}\right) \cap \left(\mathcal{T}^{[j]} \setminus \mathcal{T}^{[j+1]}\right) \\ &= \mathcal{T}^{[i]} \cap \left(\mathcal{T}^{[i+1]}\right)^C \cap \mathcal{T}^{[j]} \cap \left(\mathcal{T}^{[j+1]}\right)^C\end{aligned}$$

The model assumes the support moves in a single direction, so, for a $j > i$, if $\mathcal{T}^{[j]}$ intersects with $\mathcal{T}^{[i]}$ then it also necessarily intersects with $\mathcal{T}^{[i+1]}$. Thus, $\mathcal{T}^{[i]} \cap \left(\mathcal{T}^{[i+1]}\right)^C \cap \mathcal{T}^{[j]} = \emptyset$.

Next we show that $\mathcal{T}^{[j]} \subseteq \mathcal{T}_{(j)} \cup \cdots \cup \mathcal{T}_{(j+\varrho-1)}$. We use the fact that $\mathcal{T}^{[j]} \cap \mathcal{T}^{[j+\varrho]} = \emptyset$.

$$\begin{aligned}\mathcal{T}^{[j]} &= [\mathcal{T}^{[j]} \cap \left(\mathcal{T}^{[j+1]}\right)^c] \cup [\mathcal{T}^{[j]} \cap \mathcal{T}^{[j+1]} \cap (\mathcal{T}^{[j+2]})^C] \cup \cdots \cup [\mathcal{T}^{[j]} \cap \cdots \cap (\mathcal{T}^{[j+\varrho]})^C] \\ &= [\mathcal{T}_{(j)}] \cup [\mathcal{T}^{[j]} \cap \mathcal{T}_{(j+1)}] \cup \cdots \cup [\mathcal{T}^{[j]} \cap \mathcal{T}^{[j+1]} \cap \cdots \cap \mathcal{T}_{(j+\varrho-1)}] \\ &\subseteq \mathcal{T}_{(j)} \cup \mathcal{T}_{(j+1)} \cup \cdots \cup \mathcal{T}_{(j+\varrho-1)}\end{aligned}$$

Notice that by construction, $\mathcal{T}^{[j]} \cap \mathcal{T}_{(j)} \neq \emptyset$. This along with the fact that the support changes at least once every β time instants and the fact that once an index is removed from the support, it does not return for the next α time instants, implies that $h(\alpha) \leq \beta$ for this choice of $\mathcal{T}_{(i)}$'s. Since $h^*(\alpha) \leq h(\alpha)$ and since $h^+ = \beta/\alpha$ we are done. ⊠

Proof of Lemma 3.3.5. For this model, we choose $\mathcal{T}_{(j)} = [(j-1)s + 1, js]$ and let $\rho = 2$. The assumption that $\alpha \leq \frac{n}{m}$ ensures that no indices re-enter the support after being removed in an interval $\mathcal{J}_u$ (which has length α). Because the support moves down by at least one index at every time t and no indices are re-visited, $h^*(\alpha) \leq s$. Therefore $h^+ = \frac{s}{\alpha}$. The assumptions then imply that $\rho^2 h^+ \leq .0024$, which satisfies Corollary 3.3.2. ⊠

3.C Bounding $\zeta^+_{k,\text{new}}$ [For the purposes of review]

Recall that $\zeta^+_{k,\text{new}} := \dfrac{b_{\boldsymbol{\mathcal{H}},k}}{b_{\boldsymbol{A}} - b_{\boldsymbol{A},\perp} - b_{\boldsymbol{\mathcal{H}},k}}$.

Let $\epsilon = 0.05 c_{\text{new}} \zeta \lambda^-$. Divide the numerator and denominator by λ^-_{new} and use the bounds $f = \frac{\lambda^+}{\lambda^-}$ and $g = \frac{\lambda^+_{\text{new}}}{\lambda^-_{\text{new}}}$ to define

$$B_k := \begin{cases} \left[\rho^2 h^+ (\phi^+)^2 (\kappa_s^+)^2 (\zeta^+_{j,\text{new},k-1}) + 2\kappa_s^+ \phi^+\right] \left(b^2 c_{\text{new}} \frac{{\gamma_{\text{new}}}^2}{\lambda^-_{\text{new}}} + (1-b^2) g\right) + \\ \left[2\rho^2 h^+ (\phi^+)^2 \zeta^+_{j,*} + \zeta^+_{j,*} \frac{2\kappa_s^+ \phi^+}{\alpha(1-b^2)}\right] \left(b^2 \sqrt{r c_{\text{new}}} \frac{\gamma \gamma_{\text{new}}}{\lambda^-_{\text{new}}}\right) & k = 1 \\ \\ \left[\rho^2 h^+ (\phi^+)^2 \zeta^+_{j,\text{new},k-1} + 2\sqrt{\rho^2 h^+} \phi^+\right] \left(b^2 c_{\text{new}} \frac{{\gamma_{\text{new}}}^2}{\lambda^-_{\text{new}}} + (1-b^2) g\right) + \\ \left[2\rho^2 h^+ (\phi^+)^2 \zeta^+_{j,*} + \zeta^+_{j,*} \frac{2\sqrt{\rho^2 h^+} \phi^+}{\alpha(1-b^2)}\right] \left(b^2 \sqrt{r c_{\text{new}}} \frac{\gamma \gamma_{\text{new}}}{\lambda^-_{\text{new}}}\right) & k \geq 2 \end{cases}$$

$$C_k := \begin{cases} \left[\rho^2 h^+ (\phi^+)^2 (\zeta^+_{j,*}) r + 2\phi^+ (\zeta^+_{j,*}) r + 2(\zeta^+_{j,*}) r\right] \left(b^2 r \frac{\gamma^2}{\lambda^-_{\text{new}}} + (1-b^2) f\right) + \\ \left[\frac{2\phi^+ r}{\alpha(1-b^2)} + \frac{2r}{\alpha(1-b^2)}\right] \left(b^2 \sqrt{r c_{\text{new}}} \frac{\gamma \gamma_{\text{new}}}{\lambda^-_{\text{new}}}\right) + 0.25 & k = 1 \\ \\ \left[\rho^2 h^+ (\phi^+)^2 (\zeta^+_{j,*}) r + 2\sqrt{\rho^2 h^+} \phi^+ (\zeta^+_{j,*}) r + 2(\zeta^+_{j,*}) r\right] \left(b^2 r \frac{\gamma^2}{\lambda^-_{\text{new}}} + (1-b^2) f\right) + \\ \left[\frac{2\sqrt{\rho^2 h^+} \phi^+ r}{\alpha(1-b^2)} + \frac{2r}{\alpha(1-b^2)}\right] \left(b^2 \sqrt{r c_{\text{new}}} \frac{\gamma \gamma_{\text{new}}}{\lambda^-_{\text{new}}}\right) + 0.25 & k \geq 2 \end{cases}$$

$$\begin{aligned} D_k :=& 1 - (\zeta^+_{j,*})^2 - b^2 - \left[2\zeta^+_{j,*} \frac{1}{\alpha(1-b^2)}\right] \left(b^2 \sqrt{r c_{\text{new}}} \frac{\gamma \gamma_{\text{new}}}{\lambda^-_{\text{new}}}\right) - \\ & \left[(\zeta^+_{j,*})^2\right] \left(b^2 r \frac{\gamma^2}{\lambda^-_{\text{new}}} + (1-b^2) f\right) - \\ & \zeta^+_{j,\text{new},k-1} B_k - c_{\text{new}} \zeta C_k - 0.15 \end{aligned}$$

Then,

$$\zeta^+_{\text{new},k} \leq \zeta^+_{\text{new},k-1} \frac{B_k}{D_k} + c_{\text{new}} \zeta \frac{C_k}{D_k}.$$

It is not difficult to see that B_k, C_k, D_k are increasing functions of $\zeta^+_{\text{new},k-1}$ and of r, f, g, and ζ. Since $\eta \geq 1$, so $\eta f \geq f$. Thus $b^2 \eta f + (1-b^2) f$ is increasing in b and so B_k, C_k, D_k are also increasing in b.

Using the bounds assumed in Theorem 3.2.15 and since $\zeta^+_{\text{new},0} = 1$, we can get that $\zeta^+_1 \leq 0.15$. Using this and the fact that B_k, C_k, D_k are increasing functions of $\zeta^+_{\text{new},k-1}$, by induction, we can show that $\zeta^+_{\text{new},k} \leq \zeta^+_{\text{new},k-1}$ and thus, for all $k \geq 1$, $\zeta^+_{\text{new},k} \leq 0.15$.

Using $\zeta^+_{\text{new},k} \leq 0.15$ and the bounds assumed in Theorem 3.2.15, we can show that

$$\zeta^+_{k,\text{new}} \leq 0.4\zeta^+_{j,\text{new},k-1} + 0.5c_{\text{new}}\zeta$$

Thus,

$$\begin{aligned}\zeta^+_{k,\text{new}} \leq 0.4\zeta^+_{j,\text{new},k-1} + 0.5c_{\text{new}}\zeta &= \zeta^+_0(0.4)^k + \sum_{i=0}^{k-1}(0.4)^i(0.5)c_{\text{new}}\zeta \\ &\leq \zeta^+_0(0.4)^k + \sum_{i=0}^{\infty}(0.4)^i(0.5)c_{\text{new}}\zeta \\ &\leq 0.4^k + 0.84c_{\text{new}}\zeta\end{aligned}$$

References

[1] J. Wright and Y. Ma, "Dense error correction via l1-minimization," *IEEE Trans. on Info. Th.*, vol. 56, no. 7, pp. 3540–3560, 2010.

[2] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of ACM*, vol. 58, no. 3, 2011.

[3] V. Chandrasekaran, S. Sanghavi, P. A. Parrilo, and A. S. Willsky, "Rank-sparsity incoherence for matrix decomposition," *SIAM Journal on Optimization*, vol. 21, 2011.

[4] Y. Chen, A. Jalali, S. Sanghavi, and C. Caramanis, "Low-rank matrix recovery from errors and erasures," in *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on.* IEEE, 2011, pp. 2313–2317.

[5] H. Xu, C. Caramanis, and S. Sanghavi, "Robust pca via outlier pursuit," *IEEE Tran. on Information Theorey*, vol. 58, no. 5, 2012.

[6] M. B. McCoy and J. A. Tropp, "Sharp recovery bounds for convex demixing, with applications," *arXiv:1205.1580*.

[7] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky, "The convex geometry of linear inverse problems," *Foundations of Computational Mathematics*, no. 6, 2012.

[8] M. Tao and X. Yuan, "Recovering low-rank and sparse components of matrices from incomplete and noisy observations," *SIAM Journal on Optimization*, vol. 21, no. 1, pp. 57–81, 2011.

[9] F. D. L. Torre and M. J. Black, "A framework for robust subspace learning," *International Journal of Computer Vision*, vol. 54, pp. 117–142, 2003.

[10] M. Mardani, G. Mateos, and G. B. Giannakis, "Dynamic anomalography: Tracking network anomalies via sparsity and low rank," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 7, no. 1, pp. 50–66, 2013.

[11] J. A. Tropp, "User-friendly tail bounds for sums of random matrices," *Foundations of Computational Mathematics*, vol. 12, no. 4, 2012.

[12] C. Qiu, N. Vaswani, B. Lois, and L. Hogben, "Recursive robust pca or recursive sparse recovery in large but structured noise," *IEEE Trans. Info. Th.*, Aug. 2014, shorter versions in ICASSP 2013 and ISIT 2013.

[13] B. Lois and N. Vaswani, "A correctness result for online robust pca," *arXiv:1409.3959*.

[14] H. Guo, C. Qiu, and N. Vaswani, "An online algorithm for separating sparse and low-dimensional signal sequences from their sum," *IEEE Trans. Sig. Proc.*, pp. 4284–4297, Aug. 2014.

[15] J. He, L. Balzano, and J. Lui, "Online robust subspace tracking from partial information," *arXiv:1109.3827 [cs.IT]*, 2011.

[16] G. Mateos and G. B. Giannakis, "Robust pca as bilinear decomposition with outlier-sparsity regularization," *Signal Processing, IEEE Transactions on*, vol. 60, no. 10, pp. 5176–5190, 2012.

[17] D. Hsu, S. Kakade, and T. Zhang, "Robust matrix decomposition with sparse corruptions," *IEEE Trans. Info. Th.*, Nov. 2011.

[18] J. Feng, H. Xu, and S. Yan, "Online robust pca via stochastic optimization," in *Advances in Neural Information Processing Systems 26*, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 404–412. [Online]. Available: http://papers.nips.cc/paper/5131-online-robust-pca-via-stochastic-optimization.pdf

[19] J. Feng, H. Xu, S. Mannor, and S. Yan, "Online pca for contaminated data," in *Advances in Neural Information Processing Systems 26*, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 764–772. [Online]. Available: http://papers.nips.cc/paper/5135-online-pca-for-contaminated-data.pdf

[20] J. Zhan and N. Vaswani, "Robust pca with partial subspace knowledge," *arXiv:1403.1591 [cs.IT]*, 2014.

[21] S. Li and H. Qi, "Recursive low-rank and sparse recovery of surveillance video using compressed sensing," in *Proceedings of the International Conference on Distributed Smart Cameras*, ser. ICDSC '14. New York, NY, USA: ACM, 2014, pp. 1:1–1:6. [Online]. Available: http://doi.acm.org/10.1145/2659021.2659029

[22] B. Nadler, "Finite sample approximation results for principal component analysis: A matrix perturbation approach," *The Annals of Statistics*, vol. 36, no. 6, 2008.

[23] E. Candes, "The restricted isometry property and its implications for compressed sensing," *Compte Rendus de l'Academie des Sciences, Paris, Serie I*, pp. 589–592, 2008.

[24] M. Muselli, "On convergence properties of pocket algorithm," *Neural Networks, IEEE Transactions on*, vol. 8, no. 3, pp. 623–629, May 1997.

[25] D. Hsu, S. M. Kakade, and T. Zhang, "Robust matrix decomposition with sparse corruptions," *Information Theory, IEEE Transactions on*, vol. 57, no. 11, pp. 7221–7234, 2011.

[26] C. Davis and W. M. Kahan, "The rotation of eigenvectors by a perturbation. iii," *SIAM Journal on Numerical Analysis*, Mar. 1970.

[27] R. Horn and C. Johnson, *Matrix Analysis*. Cambridge Univ. Press, 1985.

CHAPTER 4. GENERAL CONCLUSIONS

In this work correctness results for two versions the ReProCS algorithm were proved. In Chapter 2, results were obtained for both the online matrix completion problem and the online robust PCA problem. It was shown that if 1) the algorithm parameters were set appropriately, 2) the $\boldsymbol{\ell}_t$ are independent over t and lie in a dense low-dimensional subspace that changes slowly over time, and 3) the support of $\boldsymbol{x}_t$ changes enough with t, then the ReProCS algorithm will accurately recover $\boldsymbol{x}_t$ and $\boldsymbol{\ell}_t$ at each time t.

Chapter 3 analyzed a modification of the ReProCS algorithm that included a cluster PCA step. This allowed for directions to be removed from the estimate of the subspace where the $\boldsymbol{\ell}_t$ lie. Also in this paper, the assumption that the $\boldsymbol{\ell}_t$ be independent was relaxed, and an autoregressive model was assumed on the coefficients $\boldsymbol{a}_t$.

In ongoing work, we have been able to remove the assumption that the eigenvalue clusters are known a priori. Using the same proof techniques (Weyl's inequality and the Matrix Hoeffding inequality), we can show that for each i, $\lambda_i\left(\sum_t \hat{\boldsymbol{\ell}}_t \hat{\boldsymbol{\ell}}_t{}'\right)$ is close to $\lambda_i\left(\sum_t \boldsymbol{\ell}_t \boldsymbol{\ell}_t{}'\right)$. Thus, the clusters can be detected automatically.

As mentioned in Chapter 2, one direction for new work would be to study the undersampled case $\boldsymbol{m}_t = \boldsymbol{A}_t \boldsymbol{x}_t + \boldsymbol{B}_t \boldsymbol{\ell}_t$ where $\boldsymbol{A}_t$ and $\boldsymbol{B}_t$ are matrices with more columns than rows. A partial result for the case where $\boldsymbol{B}_t$ does not change with t was proved in [1]. Using the new techniques introduced in this thesis, it should be straightforward to extend this result to a complete correctness result. Another open problem is the noisy case: $\boldsymbol{m}_t = \boldsymbol{x}_t + \boldsymbol{\ell}_t + \boldsymbol{w}_t$ where $\boldsymbol{w}_t$ is small and bounded noise.

References

[1] B. Lois, N. Vaswani, and C. Qiu, "Performance guarantees for undersampled recursive sparse recovery in large but structured noise," in *GlobalSIP*, 2013.